DIGITAL ACCESS TO SCHOLARSHIP AT HARVARD
DASH.HARVARD.EDU

HARVARD LIBRARY
Office for Scholarly Communication

Symmetry Breaking in Neuronal Development

Citation

Wissner-Gross, Zachary Daniel. 2012. Symmetry Breaking in Neuronal Development. Doctoral dissertation, Harvard University.

Permanent link

http://nrs.harvard.edu/urn-3:HUL.InstRepos:9824174

Terms of Use

This article was downloaded from Harvard University's DASH repository, and is made available under the terms and conditions applicable to Other Posted Material, as set forth at http://nrs.harvard.edu/urn-3:HUL.InstRepos:dash.current.terms-of-use#LAA

Share Your Story

The Harvard community has made this article openly available.
Please share how this access benefits you. Submit a story .

Accessibility

© 2012 -- Zachary Wissner-Gross
All rights reserved.

Professor Mehmet Fatih Yanik Zachary Wissner-Gross

Symmetry Breaking in Neuronal Development

Abstract

Many physical systems break symmetry in their evolution. Biophysical systems, such as cells, developing organisms, and even entire populations, are no exception. Developing neurons represent a striking example of a biophysical system that breaks symmetry: neurons cultured *in vitro* begin as cell bodies with several tendrils ("neurites") growing outward. A few days later, these same neurons invariably have the same new morphology: exactly one of the neurites (the "axon") has grown hundreds of microns in length, while the others (the "dendrites") are much shorter and are more branched. Previous work has shown that any of the neurites can become the axon, and so neurons must break symmetry during their development. The mechanisms underlying neuronal symmetry breaking and axon specification have recently attracted attention, with multiple groups proposing biophysical models to explain the phenomena.

In this thesis, we perform the first analytical comparisons of these models by conducting multiple phenotypic and morphological studies of neurite growth in developing neurons. Studying neurite dynamics is technically challenging because neurites have unpredictable morphologies. In Chapter 4, we study neurite competition and neuronal symmetry breaking in hundreds of neurons by optically patterning micron-wide stripes to which the neurons adhere, and on which they grow exactly two neurites. We then use our measurements to test the accuracy of the models in the simple case when a neuron has exactly two neurites.

In Chapter 5, we no longer constrain neuronal morphology. One characteristic of symmetry breaking systems is how the system's complexity affects the symmetry breaking. We find that a majority of the models predict that neurons with more neurites break symmetry much slower than neurons with fewer neurites. Experimentally, we find that neurons with different neurite counts break symmetry at the same rate, consistent with previous observations. We then determine why the models disagree in their

predictions, and rectify the models using our own experimental data. In particular, we find that neurons with higher neurite counts have higher concentrations of key proteins involved in symmetry breaking, so that neurons, regardless of neurite count, can break symmetry at the same rate.

Table of Contents

Acknowledgments

First, I would like to thank the members of the Yanik Lab. My partner in constructing the laser patterning apparatus, Mark Scott, has been a great friend over the last five years and he has also done a wonderful job helping to maintain the system since we first built it. I also want to specially thank Mark, Peng Shi, and Joe Steinmeyer for their willingness to harvest primary neurons over the years. I also thank the undergraduate and high school students whom I've had the pleasure of mentoring and collaborating: David Ku, Eulalia Nicolau, Liza Zheng, and Priya Ramaswamy. Finally, I want to thank Mark, Joe, Cody Gilleland, Chris Rohde, Matt Angel, Carlos Pardo, David Ku, Chrysa Samara, and Janice Balzer for their years of help and support around the lab.

I am also very grateful to Prof. M. Fatih Yanik for his guidance and supervision. His lab has been an exciting place over the years, and I have learned so much from the great people who work in it. I want to thank the rest of my thesis committee as well: Prof. Andrew Kiruluta, Prof. Joao Guimaraes da Costa, and especially Prof. Erel Levine, for their conversations and thoughtful suggestions over the years. I am also grateful to the Harvard Physics staff, especially Sheila Ferguson and Lisa Cacciabaudo.

Next, I would like to thank the inspiring math and science teachers I've had over the years, who helped set me on a path of exploration and learning: Matthew Sckalor, Joyce O'Connor, Rose Grabelsky, Eileen Magilligan, Walter Lewin, David Morin, and Young Lee.

I would like to thank my friends who have helped me through my tenure in grad school. In particular, I thank Vivek Venkatachalam for his constant advice, as well as my many friends who kept me sane over the years, inlcuding Javier Matamoros, Erin Koksal, John Lee, Jen Balakrishnan, and Alex Hayman. And I especially want to thank Randi Goldman for her unwavering love and support.

My work was supported by both a Fannie and John Hertz Foundation/Myhrvold Family Fellowship, as well as by a National Defense Science and Engineering Graduate (NDSEG) Fellowship from the Department of Defense.

Finally, I would like to thank my family: my grandparents Aileen Wissner, Cynthia Gross, and Robert Gross, as well as Artie Nussbaum, for their encouragement (and food!); my brother Alex Wissner-Gross, along with Tim Sullivan, for offering advice whenever I needed it, and for helping me get into

Harvard Physics in the first place (they did, after wall, wake up at an ungodly hour to drive me to Salem, MA, to take the Physics GRE). Last, I would like to thank my parents, who have been there every step of the way, from taking me to science museums, to cheering me on at math competitions, science fairs, fencing tournaments, and musical productions, to even proofreading this thesis. They are, without question, the best parents one could ever have.

"What God giveth, symmetry taketh away."

— Harvard Physics G2 Puppet Show

Chapter 1

Symmetry breaking in developing neurons

In this chapter we first describe several examples of symmetry breaking in biophysical systems. We then specifically discuss the developing neuron, a system whose symmetry breaking has been actively explored in recent years. Finally, we introduce several biophysical models, with varying complexity, that have been created to explain neuronal symmetry breaking.

1.1 Symmetry breaking in biophysics

The notion of spontaneous symmetry breaking, or the sudden transition from a symmetric state to one that is asymmetric, is nothing new to physics. In quantum mechanics, the symmetry of wave functions can be spontaneously broken when they are collapsed by measurement. In particle physics, spontaneous symmetry breaking is an essential component of the Standard Model, explaining why W and Z bosons have mass. And in cosmology, the anisotropy of the cosmic microwave background could represent symmetry breaking in the early Universe.

Biological systems also commonly undergo spontaneous symmetry breaking events. In this thesis, we go into great detail of how neurons break symmetry during development, but other cell types, such as motile cells (e.g., leukocytes) polarize in isotropic environments as well [1]. Other examples of biological symmetry breaking include asymmetric cell division and morphogenesis (fetuses, after all, must break symmetry along all three spatial axes as they develop *in utero*) [2].

One principle that underlies spontaneous symmetry breaking in biological systems is a biophysical or biochemical amplification mechanism, so that relatively small asymmetries can quickly become system-wide, locking the cell (or tissue, organism, etc.) into an asymmetric state. This is analogous to the "Mexican hat potential," a two-dimensional Gaussian-shaped potential (resembling a sombrero), in which a

particle placed symmetrically at the peak of the function is unstable. It is energetically favorable for the particle to break symmetry and roll off to some side. As it rolls, it gains kinetic energy, and continues moving in that direction, and so the potential represents an amplification or "positive feedback" mechanism for symmetry breaking.

A number of symmetry breaking mechanisms with positive feedback have been proposed to explain how developing neurons break symmetry, and we will review them in Section 1.3. First, in Section 1.2, we will further discuss recent experimental findings in neuronal symmetry breaking.

1.2 Symmetry breaking in developing neurons

When neurons develop *in vitro*, several minor processes, called "neurites," grow out of the cell body (Figure 1.1, Stages 1 and 2) [3]. After 24-48 h, one of the neurites begins to grow faster than the others (Stage 3), and this one ultimately becomes the neuron's single axon (Stage 4). After about a week in culture, the other neurites gradually mature to form neurites (Stage 5). However, from this picture of neuronal development, it is not clear whether symmetry is broken between Stages 2 and 3, or whether the axon is somehow specified at an earlier time (or even once the neuron is first formed) but does not become structurally prominent until Stage 3.

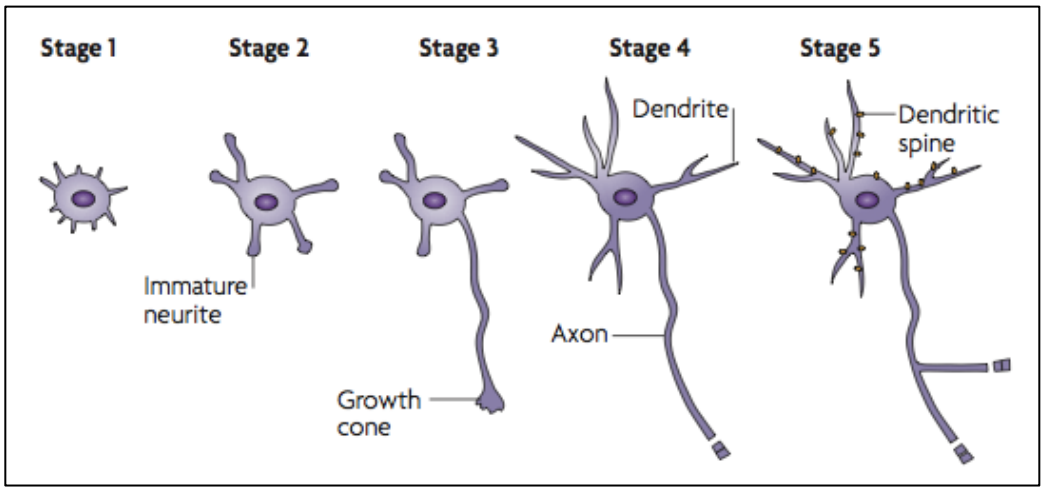

Figure 1.1. Stages of neuronal development. The breaks in the axon in Stages 4 and 5 are meant to signify that the axon grows very long (up to hundreds of microns, as compared to the cell body, which is typically ~20 μm in diameter). Image taken from [4].

Axons and dendrites are very different, both structurally and functionally. Axons are much longer structures, can be myelinated, and action potentials travel unidirectionally down axons from one neuron to the next. Dendrites are typically much shorter, and integrate signals from the axons of other (or sometimes the same) neurons. The junctions between axon and dendrite are called synapses.

Despite these differences between axons and dendrites, Banker and colleagues published several papers around 1990 showing that each of the neurites (which can vary in number, typically between 3 and 9) possesses the ability to become the axon – that axon specification is not predetermined in the neuron [3,5,6]. Banker's team determined this by axotomizing (i.e., cutting the axon of) polarized Stage 3 neurons *in vitro*, only to find that another neurite could grow out and ultimately become the axon. Gosline and Banker also found that in post-axotomized neurons, the longest neurite (even if it was the one that had been cut) tended to become the axon [6]. Thus, during development, neurons progress from a symmetric state to one that is highly asymmetric. Neuronal development and axon specification therefore represent an excellent example of spontaneous symmetry breaking in a biological system.

Over the past 20 years, there have been several efforts to determine the precise biophysical mechanism underlying neuronal symmetry breaking. Figure 1.2, summarizes the two leading hypotheses: positive regulation (i.e., a particle rolling faster and faster down the Mexican hat potential) and negative regulation (analogous to the particle making a depression in the potential, constraining the particle to continue breaking symmetry in that direction, but not in the other directions). In positive regulation/feedback, longer or more axonal neurites further induce their own growth, either through membrate recruitment, protein transport, increased F-actin dynamics, and/or microtubule assembly [4]. Negative regulation/feedback then describes when neurites have the direct opposite effect on competing neurites. Positive feedback is sufficient for axon specification; indeed, all the biophysical models described in Section 1.3 lack a negative feedback term. Instead, the neurites compete for a common scarce resource necessary for neurite growth – longer neurites accumulate more of this resource through a positive feedback term, while shorter neurites do not.

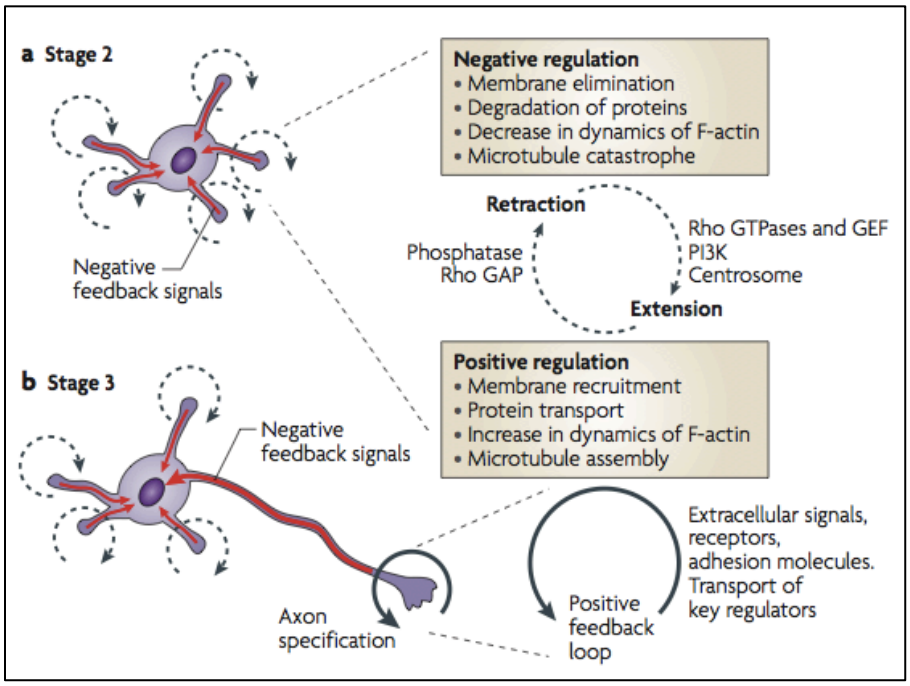

Figure 1.2. Hypothetical positive and negative regulation/feedback in neurons as they break symmetry and transition from Stage 2 to Stage 3. Image taken from [4].

The models differ in the positive feedback loops they identify as being responsible for neuronal symmetry breaking. This is not particularly surprising, given the complexity of the molecular pathways involved (Figure 1.3). In the next section, we introduce these models.

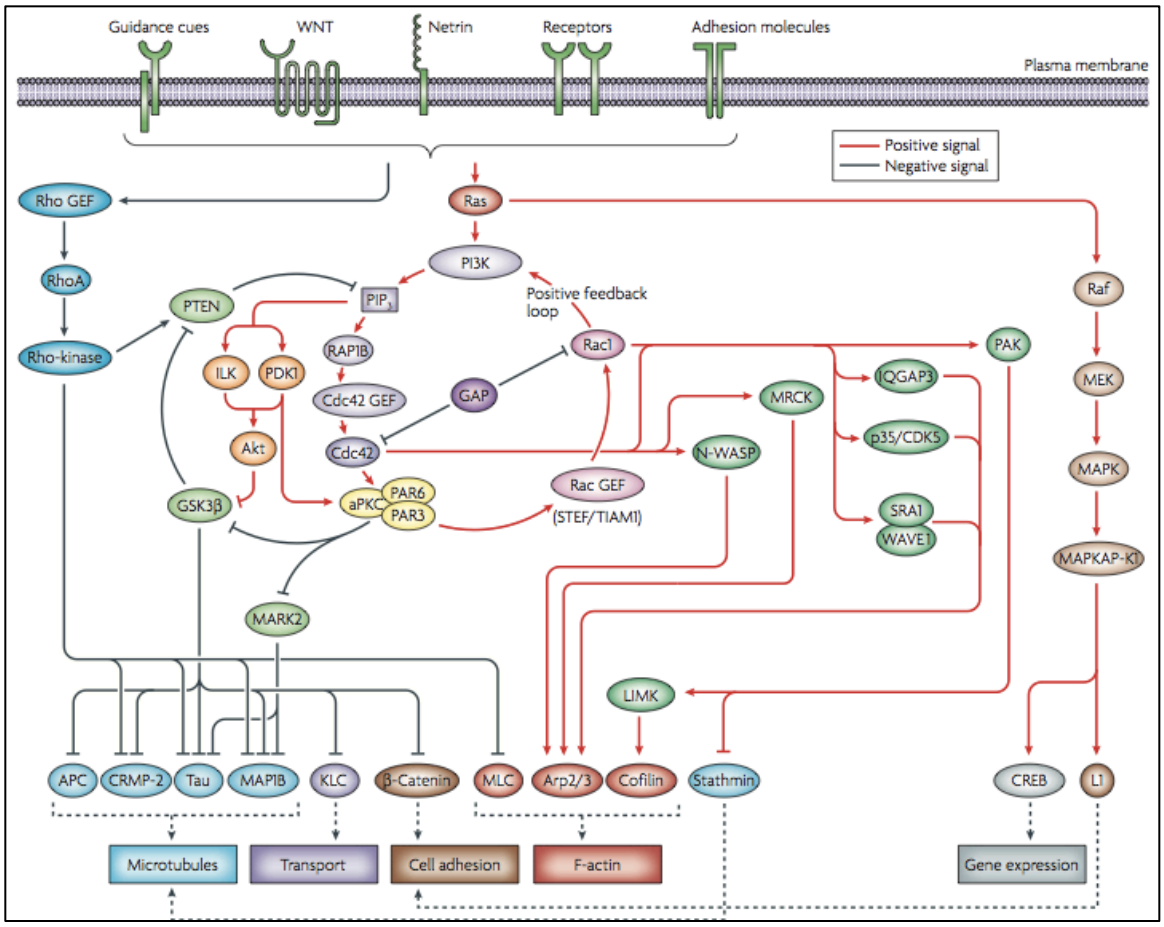

Figure 1.3. Molecular pathways involve in neurite outgrowth and neuronal symmetry breaking. Image taken from [4].

1.3 Models for neuronal symmetry breaking

In this section, we review several of the leading models for neuronal symmetry breaking, which we analyze further in Chapters 4 and 5. Each of the models assumes that neurite count is fixed throughout polarization, an assumption we disprove in Chapter 5. MATLAB code for the Khanin, Samuels, Fivaz, and Toriyama models can be found in Appendix A.

1.3.1 Khanin model

The Khanin model was formulated as a simple mathematical model capable of recapitulating axon specification [7]. The model assumes that at fixed time intervals, exactly one of a neuron's N neurites can

grow an additional unit of length, and that the probability of any neurite growing relates to that neurite's current length via a power law. In the Khanin model, the probability of neurite i (with corresponding length L_i) growing an additional unit at a specific time interval is:

$$p\left(L_i \rightarrow L_i + 1\right) = \frac{L_i^{\alpha}}{\sum_{j=1}^{N} L_j^{\alpha}}. \tag{1.1}$$

The denominator ensures that the probabilities are normalized, and the exponent α represents the strength of the competition between the neurites. Khanin et al. analytically prove that for $\alpha > 1$, the neuron is guaranteed to eventually polarize (i.e., beyond some time T, one neurite will grow at every time step, while all the other neurites will cease to grow), while the neuron will not polarize for $\alpha \leq 1$ [7].

While the original Khanin model makes use of randomness, in Chapter 4 we compare experimental data to a deterministic version of the model, in order to more easily measure α. The deterministic Khanin model can be written as:

$$\dot{L}_i = v_0 \frac{L_i^{\alpha}}{\sum_{j=1}^{N} L_j^{\alpha}}. \tag{1.2}$$

In this version of the model, α continues to represent strength of the competition between the neurites, but v_0 is an additional parameter, representing the steady-state growth rate of the axon once it has been specified. In Chapter 4, using thousands of data points, we are able to determine the values of α and v_0 for hippocampal neurons *in vitro*.

1.3.2 Samuels model

The Samuels model is similar to the Khanin model in that it does not model specific proteins, but instead a generic chemical that is rate-limiting for neurite growth [8]. But in addition to a positive feedback term, the Samuels model also takes into account the diffusion of this chemical between the cell body and the neuron's N neurites. In the model, the concentration of the chemical in the cell body is C_0, and the concentration in the tip of neurite i is C_i (Figure 1.4). The length of neurite i is represented as L_i.

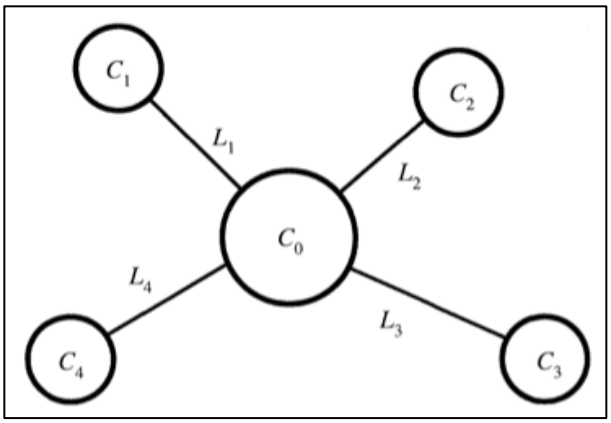

Figure 1.4. Schematic of a neuron in the Samuels model with four neurites. C_0 represents the concentration of the rate-limiting chemical for neurite growth in the cell body, C_i are the concentrations in the neurite tips, and L_i are the lengths of the neurites. Figure taken from [8].

The non-dimensionalized equations of the Samuels model are then

$$\dot{C}_0 = 1 - \sum_{i=1}^{N} \left(\chi_1 \frac{C_0 - C_i}{L_i} + \chi_2 C_0 \dot{L}_i \right) \tag{1.3}$$

$$\dot{C}_i = \chi_3 \left(\chi_1 \frac{C_0 - C_i}{L_i} + \chi_2 C_0 \dot{L}_i - C_i \right) \tag{1.4}$$

$$\dot{L}_i = C_i \,. \tag{1.5}$$

The chemical diffuses back and forth between the cell body and neurite tips (governed by χ_1), and the faster a neurite grows, the more of the chemical it recruits (governed by χ_2). The chemical is consumed as the neurite grows, and a neurite's growth rate is assumed to be proportional to the concentration of the chemical in that neurite's tip. In addition to the three dimensionless parameters χ_1, χ_2, and χ_3, time and length are scaled by a characteristic time t_{scale} and length L_{scale}. In Chapter 4, as with the Khanin model, we experimentally measure values for these five parameters.

Neurons are able to polarize in the Samuels model because as they grow longer, the diffusion of the chemical back to the cell body slows down. A similar mechanism is employed in the Toriyama model (see Section 1.3.4).

1.3.2.1 Analysis of the Samuels model

An intriguing question posed by Samuels et al. is for which values of χ_1, χ_2, and χ_3 neurons will polarize. They found that polarization behavior depended only weakly on χ_3, and they produced Figure 1.5 in their original paper. While Samuels et al. offer no analytical justification for the shape of the curve they found computationally, we find that axons cannot form when $\chi_2 < N$, consistent with Figure 1.5.

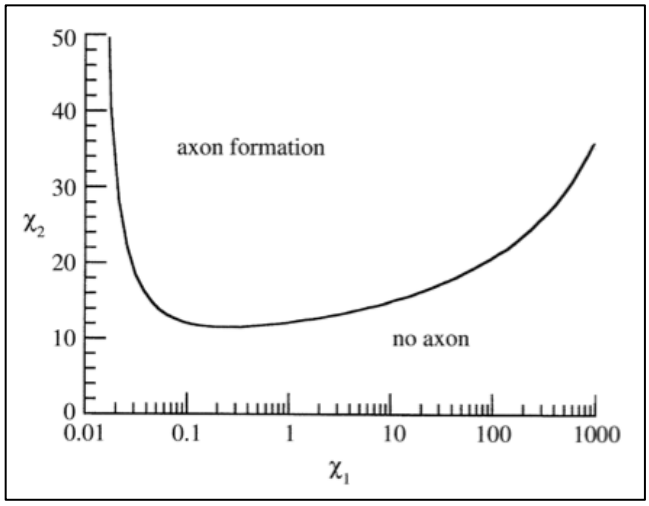

Figure 1.5. Regimes in the χ_1, χ_2 parameter space in which neurons do or do not polarize under the Samuels model for $N = 4$. Figure taken from [8], and compare with Figure 1.6.

We first consider the steady state limit regardless of whether an axon has formed or not, and at least one neurite i has grown to normalized length $L_i >> 1$. In this case, we can neglect the first term in the parentheses in Eq. (1.4), and setting the derivative of the concentration equal to zero for neurite i, we find that

$$C_0 = \frac{1}{\chi_2}. \tag{1.6}$$

Thus, in the steady-state limit, we know the concentration in the cell body.

Next, we can determine the concentrations in the neurite tips in the steady-state limit when there is no axon. As the neurites all grow very long symmetrically, we can ignore the first term inside the sum of Eq. (1.3), and so in the steady-state limit:

$$0 = 1 - \chi_2 C_0 \sum_{i=1}^{N} C_i \tag{1.7}$$

Combining Eqs. (1.6) and (1.7), and again assuming in the symmetric case that C_i is the same for all i, we find that

$$C_i = \frac{1}{N}. \tag{1.8}$$

Now let us consider the case when $\chi_2 < N$ and the C_i's are all near $1/N$, so that $C_i < C_0$ for all i. In this case, diffusion flows forward to the neurites (the opposite of what typically occurs in the Samuels model), so that shorter neurites are now at an advantage. Thus, when $\chi_2 < N$, a key symmetry breaking element (less diffusion back to the cell body in longer neurites) breaks down, and symmetric neurons remain symmetric. In other words, when $\chi_2 < N$, neurons in the Samuels model cannot break symmetry, and axons will not form.

1.3.2.2 A stochastic Samuels model

The Samuels model differs from the Khanin, Fivaz, and Toriyama models in that it is entirely deterministic. Inspired by these other models, we decided to test the effect of randomness on the Samuels model by introducing a stochastic function that we multiplied by the anterograde transport terms in Eqs. (1.3) and (1.4), effectively condensing the chemical during anterograde transport into Gaussian pulses occurring at random times. We defined the dimensionless parameter J as being the ratio of the pulse width to the mean inter-pulse spacing. We were also careful not to change the total amount of chemical being transported, i.e., the amplitude of the Gaussian pulses function were set so that the entire function averaged to zero.

The effect of J on axon formation in the χ_1, χ_2 parameter space is shown in Figure 1.7 (red indicates significant axon formation, while blue indicates very little axon formation). The original Samuels model is equivalent to the stochastic model in the limit of large J, when the width of the pulses exceeds the inter-pulse spacing so that the stochastic function approaches the identity function. But as J increases, we find axon formation drops off for lower values of χ_1 (i.e., when diffusion is very weak).

With the introduction of a single additional parameter that represents a general biological behavior (random transport), in keeping with the original Samuels model, we nevertheless find that we can significantly alter the behavior of the model.

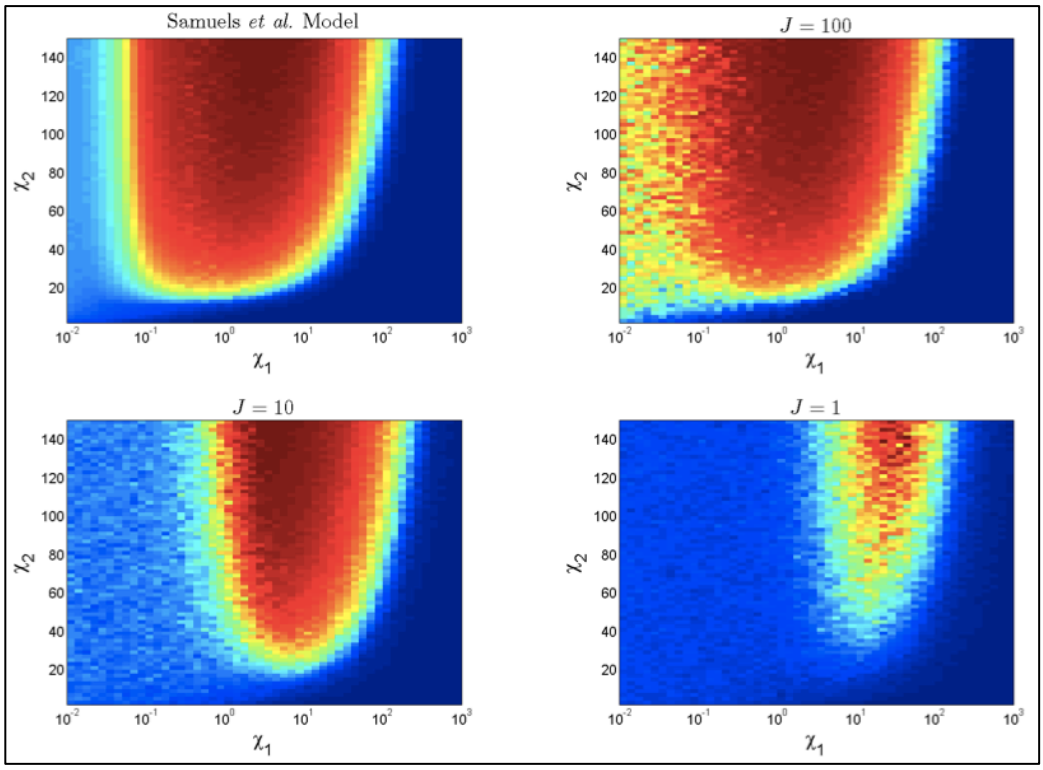

Figure 1.6. Regimes in the χ_1, χ_2 parameter space in which neurons do or not polarize under the original Samuels model (top left) and for the stochastic Samuels model we developed, for $J = 100$, 10, and 1, when $N = 4$. In the limit of large J, the stochastic model approaches the behavior of the original Samuels model. In these graphs, polarity was calculated as the relative length of the neuron's longest neurite over multiple Monte Carlo simulations. Figure created with Eulalia Nicolau.

1.3.3 Fivaz model

In 2008, Fivaz et al. used fluorescence resonance energy transfer (FRET) to identify a positive feedback loop that is relatively upstream in neuronal symmetry breaking [9]. They found that the GTPase HRas, a regulator of phosphatidylinositol-3-kinase (PI3K), increases in the immature neurite that becomes the axon. The HRas induces in a local increase in PI3K, which in turn recruits more HRas from the cell body that quickly travels to the neurite tip via anterograde vesicular transport. In the Fivaz model, HRas acts as the resource (predicted by the Khanin and Samuels models) that is shared by all the neurites, and depleted by the neurite that becomes the axon.

Interestingly, the equations for the Fivaz model are independent of neurite length, meaning that the model does not take into account the diffusion of any proteins or other molecules back to the cell body. It was this diffusion that is responsible for symmetry breaking in the Samuels model, as well as in the Toriyama model (see Section 1.3.4). The Fivaz model instead relies on the feedback loop between HRas and PI3K to produce axons, although multiple axons can form if the model is overloaded with HRas. Also, like the Fivaz and Toriyama models, the Fivaz model incorporates randomness, specifically in the vesicle transport timing, similar to our modification to the Samuels model in Section 1.3.2.2.

Here are the equations for the Fivaz model:

$$\dot{P}_i = \frac{k_P\left(T_i + R_b\right)}{K_M + T_i + R_b} - \varphi P_i \tag{1.9}$$

$$\dot{T}_i = k_{R1} D_i\left(P_i + P_b\right) - \left(\rho + \beta\right) T_i \tag{1.10}$$

$$\dot{D}_i = k_{R0} S\left(P_i + P_b\right) - \beta D_i + \rho T_i - k_{R1} D_i\left(P_i + P_b\right) \tag{1.11}$$

$$\dot{S} = \beta \sum_{i=1}^{N}\left(D_i + T_i\right) - k_{R0} S\left(N P_b + \sum_{i=1}^{N} P_i\right) \tag{1.12}$$

$$\dot{L}_i = k_L\left(P_i + P_b\right) - \frac{\lambda L_i}{K_L + L_i} \tag{1.13}$$

In these equations, N again represents the neurite count, and P_i, T_i, and D_i respectively indicate the concentrations of phosphatidylinositol (3,4,5)-triphosphate (PIP_3), HRas-GTP, and HRas-GDP in the tip of neurite i. S is the somatic concentration (i.e., the concentration in the cell body) of HRas-GDP. Values for all the rate constants in the model are given by Fivaz et al [9]. In Eq. (1.13), Fivaz et al. offer no numerical values for λ the rate of neurite retraction due to tension, and because neurite length does not feed back into the other equations, we neglect this term in our later analysis of the Fivaz model. An illustration of the Fivaz model can be found in Figure 4.1C.

These are the equations given by Fivaz et al. However, they do not represent the complete Fivaz model, as the model also includes random vesicle transport. The full Fivaz model can be found in Eqs. (4.6) and (4.7).

Another difference between the Fivaz model and the Samuels and Toriyama models is that the Fivaz model lacks a production term, analogous to the "1" on the right-hand-side of Eq. (1.3). In the Fivaz model, the total amount of HRas in the neuron is fixed, and Fivaz et al. specify that each neurite should be initialized with the same HRas concentration, so that the total amount of HRas expression in the neuron is proportional to *N*. This feature of the Fivaz model induces some different behaviors in this model when neurite count is varied, as we discuss in Chapter 5.

1.3.4 Toriyama model

Like the Fivaz et al., Toriyama et al. based their model on their own experimental findings [10–12]. Toriyama et al. idenfity the protein Shootin1 as being the upstream regulator of neuronal symmetry breaking [10,11]. And as in the Samuels model, the Toriyama model explains symmetry breaking as the combined effect of shootin1 promoting neurite growth, and decreased diffusion of shootin1 from the neurite tips back to the cell body in longer neurites. The complete list of equations and parameters for the Toriyama model, as described in [11], can be found below, while a complete working version of the model (along with the other models discussed in this section) can be found in Appendix A.

$$\frac{dC_i}{dt} = -\frac{AD}{VL_i}\left(C_i - C_0\right) + w(t) \tag{1.14}$$

$$\frac{dL_i}{dt} = \delta k_{\text{on}} M - \delta k_{\text{off}} \exp\left[-\kappa f_s\left(C_i\right) + \kappa f_l\left(L_i\right)\right] \tag{1.15}$$

$$\kappa f_s\left(C_i\right) = \frac{a_s C_i^h}{K_s^h + C_i^h} \tag{1.16}$$

$$\kappa f_l\left(L_i\right) = \frac{a_l \ln\left(L_i / L_0\right)}{\ln\left(K_l / L_0\right) + \ln\left(L_i / L_0\right)} \tag{1.17}$$

$$C_0 = \frac{1.9}{1 + \left(t / 3199\right)^{-2.5}} + 0.07 - \frac{V}{V_0}\sum_{i=1}^{N} C_i \tag{1.18}$$

$$M = \frac{2.9}{1 + \left(t / 4480\right)^{-2.5}} + 1.1 - \rho\sum_{i=1}^{N} L_i \tag{1.19}$$

$$w(t) = \frac{V_w}{V} w_{\text{avg}} \sum_j a_j g\left(t - t_j\right) \tag{1.20}$$

$$g(t) = \frac{1}{\sqrt{2\pi}s} \exp\left(-\frac{t^2}{2s^2}\right) \tag{1.21}$$

Some of the equations in the original manuscript of Toriyama et al. were combined, resulting in fewer total equations. Similar to the Samuels model, C_0, C_i, and L_i respectively represent the concentration of shootin1 in the cell body, the concentration in the tip of neurite i, and the length of neurite i. While the active transport of shootin1 is modeled as Gaussian pulses (as in the Fivaz model), the amplitudes of the pulses (a_j) and the times the pulses are transmitted down the neurite shaft (t_j) are modeled using gamma distributions [11].

In Eq. (1.19), we found that the numerator of 1.9 was not sufficient for axon specification, and so we increased this number by approximately 50% for our simulations of the Toriyama model (see Section 5.4.3).

1.3.5 Other models

In Chapters 4 and 5, we study the Khanin, Samuels, Fivaz, and Toriyama models because of their range of complexity, and in the case of the Fivaz and Toriyama models, because of their molecular and biophysical detail.

Other models include the those of Graham and coworkers, who modeled neurite outgrowth and symmetry breaking using a cellular automata approach, but their transport equations are identical to those of the Samuels model [13]. The same team also previously proposed a model resembling Samuels et al., but specific to tubulin transport, and containing a few additional rate constants [14], based on previous work on microtubule dynamics [15]. Tsaneva-Atanasova et al. have further modeled neurite growth as the interaction of microtubules and actin networks with vesicle transport [16]. Li and Qin published another model for neurite growth based exclusively on traction and tensile forces [17].

Chapter 2

Laser patterning of arbitrary protein gradients

In this chapter, we first review current technologies for patterning proteins on two-dimensional surfaces. We then describe the LAMP (Laser-Assisted Micropatterning of Proteins) system that we developed, which is capable of producing arbitrary protein gradients with unprecedented speed.

2.1 Techniques for protein patterning

Numerous techniques exist for patterning functional proteins onto two- and three-dimensional surfaces. Proteins can be adsorbed onto surfaces by flowing them through microfluidic channels. Whitesides and coworkers demonstrated this technology when they used microfluidic circuits to generate laminin gradients capable of orienting neuronal polarity [18]. Hoffman-Kim and colleagues expanded this work, creating linear gradients with multiple proteins [19].

Microfluidically patterned substrates, however, result in a relatively poor spatial resolution, and are very difficult to pattern along more than one axis. Printing droplets of solutions containing different proteins or concentrations allows the patterning of arbitrary two-dimensional patterns, but suffers from poor spatial resolution [20,21].

Microcontact printing and photolithographic techniques, on the other hand, achieve a superior resolution, on the scale of several microns, and can be used to create arbitrary two-dimensional patterns. Withers et al. were able to photolithographically pattern intersecting lines of poly-D-lysine 2-5 μm wide in a study of how axons behaved at intersections of the lines [22]. Philipsborn et al., on the other hand, detail a method for creating discrete "gradients" (i.e., printed dots with spatially varying separations) of axonal guidance cues [23,24]. These technologies involve high-resolution "stamps" or "masks," which can be difficult to align when patterning multiple proteins. Moreover, they are unable to create continuous gradients, resorting to the printing of discrete patterns (e.g., circles or squares), which themselves consist of a uniform protein density, but form a gradient by spatial variation in size or separation [23].

Optical patterning overcomes the difficulties of these other techniques. West and colleagues have been able to incorporate multiple peptides into photolabile hydrogels using lasers [25–27], while Shoichet and coworkers have similarly created hydrogels functionalized with small peptides, capable of primary rat neurons [28,29]. While patterns in these studies were on the scale of 10 μm, optical patterning can achieve micron or even sub-micron resolution. Kaehr et al. used multiphoton excitation to create bovine serum albumin (BSA) cables with submicron diameters that could "corral" neurite growth [30].

Costantino and coworkers used LAPAP (Laser-Assisted Protein Adsorption by Photobleaching) to produce peptide gradients with micron-level precision [31], building upon the work of Holden et al. [32]. In LAPAP, a laser beam is used to photobleach biotin-4-fluorescein, creating a free radical that can bind to a subtstrate (e.g., BSA adsorbed to a glass surface). Biotin-streptavidin chemistry is then used to bind desired proteins to the patterned surface.

While LAPAP is a very exciting technology, its greatest drawback to date has been its slow patterning speed. To achieve significant protein density in their patterns, Costantino and coworkers scan the focal point of their laser at speed on the order of 50 μm/s. In the next section, we describe a similar setup we developed, which achieved scanning speeds of up to 1000 μm/s. Further optimizing of the scanning speed, as described in Scott, Wissner-Gross, et al., we achieved speeds in excess of 10,000 μm/s [33].

2.2 Laser patterning of proteins

2.2.1 Optical setup

While Costantino and coworkers photobleached B4F using a 473 nm laser, we instead used a 3W Ti:Sapphire laser (Spectra-Physics) operating at 780 nm (Figure 2.1). The fluorescein molecule has a large two-photon cross-section at this wavelength [34], allowing us to achieve higher patterning speeds.

An electro-optic modulator (EOM) operating at 80 MHz was used to control beam intensity (i.e., pattern density), and galvanic scanning mirrors and a motorized stage were used to create arbitrary pattern geometries in different locations on the subtrate. Additional details of the optical setup can be found in Section 4.2.

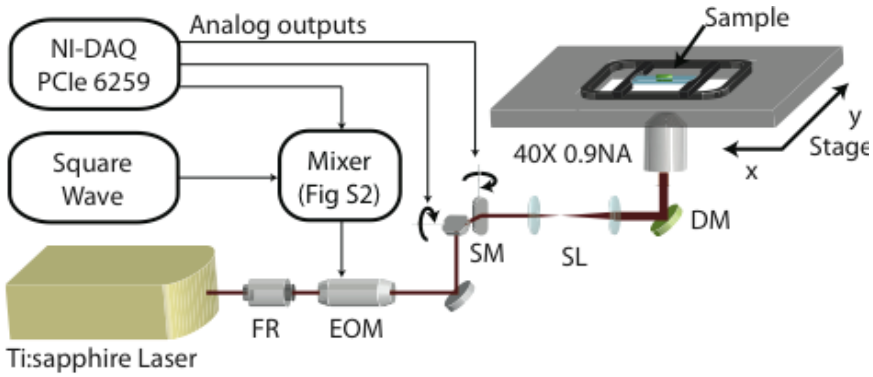

Figure 2.1. Optical setup. The Ti:sapphire laser beam (2.8 W, 780 nm) passes through an electro-optic modulator (EOM), which was used to control pattern brightness and create gradients. Position of the laser beam on the stage was controlled using two orthogonal scanning mirrors (SM) and by translating the stage itself. The beam passed through two scanning telescopic lenses (SL) that magnified the beam so it filled the back aperture of the objective (40X, 0.9 NA) after being redirected by a dichroic mirror (DM). The beam is focused by the objective onto the top surface of a PEG monolayer in contact with a fluorescein of B4F solution. Image taken from [33], and designed by Mark Scott.

2.2.2 Fluidic setup and surface chemistry

B4F, fluorescein, and other solutions were delivered to the wells of glass-bottom 96-well plates using a pneumatic delivery system. The glass surfaces are first functionalized using poly(ethylene glycol) (PEG), as described in Section 4.2. Between solutions, phosphate buffered saline (PBS) was used to rinse the wells.

The ends of the pneumatically driven fluid lines, in addition to two vacuum lines, were attached to hollow steel pins, which were held together with cured PDMS and electrical tape and mounted onto a pneumatically powered piston. These pins could be automatically lowered into any given well by moving the microscope state to the appropriate well and then lowering the piston.

2.2.3 Software

We designed and wrote our own custom program (LAMP, or Laser-Assisted Micropatterning of Proteins) for controlling both the optical and fluidic setups. Using LAMP, we were able to quickly design arbitrarily large arrays of protein patterns that uniformly varied among several geometric parameters (e.g., size, shape, complexity, etc.) and/or chemical parameters (e.g., composition, concentration, etc.).

The key features of the LAMP graphical user interface (GUI) were the main menu (Figure 2.2) and the edit_well menu (Figure 2.3). From the main manu, users can access the entire optical and fluidic setup for debugging purposes, instruct the program which fluid lines were in use and what solutions they contained, align the optical and fluidic systems, see which wells of the 96-well plate were being used, and execute specific patterning routines.

Inside the edit_well menu, users can create geometrically and chemically arbitrary protein patterns. Figure 2.3 shows one of the more complex designs, "bifurcted gradients," which can be used to study neurite growth along geometrically symmetric tracks with different protein compositions. The edit_well menu also offers a constantly updated estimate of how long the programmed routine will take to pattern.

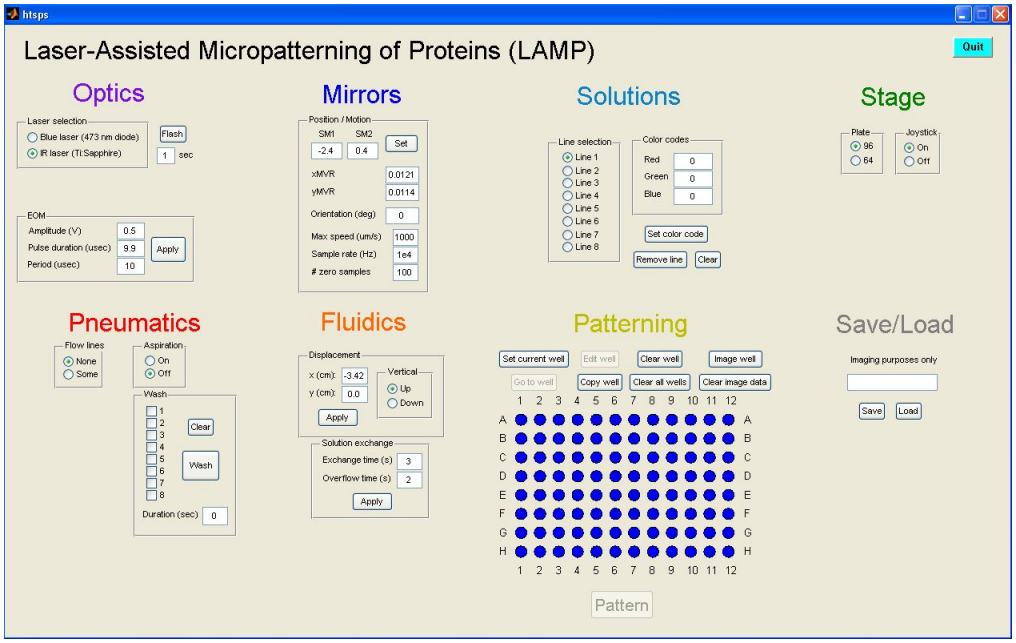

Figure 2.2. LAMP main menu. From here, users can troubleshoot the optical and fluidic systems, program in which fluid lines contain which solutions, and access the edit_well menu for any well of a 96-well plate.

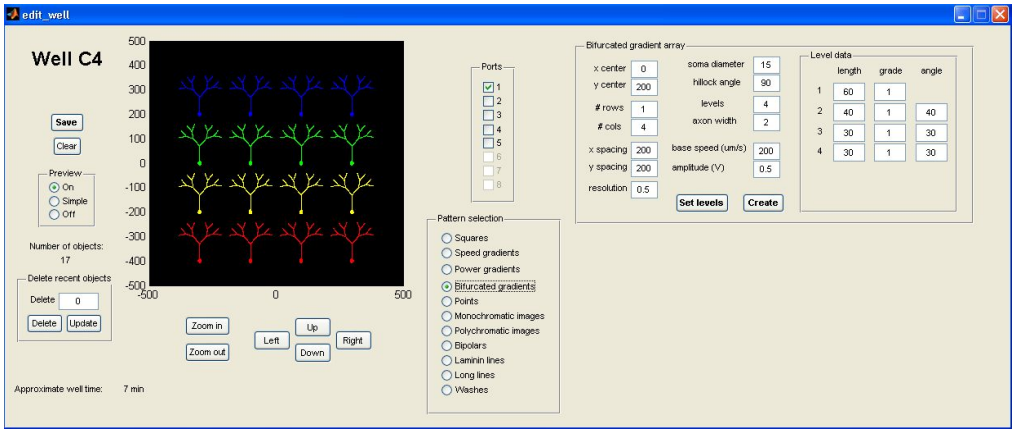

Figure 2.3. edit_well menu. From this window, users can program in arrays of any size, with any geometry, and consisting of any proteins that have been loaded into the fluidic chambers of the system. From the main menu, users can then print these patterns.

2.2.4 Creating arbitrary protein patterns

Before patterning proteins for biophysical studies of neuron development (as described in Chapter 4), we first tested the capabilities of our patterning system. Figure 2.4 shows that we were able to make lines with arbitrary gradients, and Figure 2.5 is one of the many micron-scale illustrations we made with the system. The gradients in Figure 2.4 are quadratic functions with different coefficients, and are composed of fluorescein, while those in Figure 2.5 consist of fluorescent streptavidin that has adhered to patterned B4F.

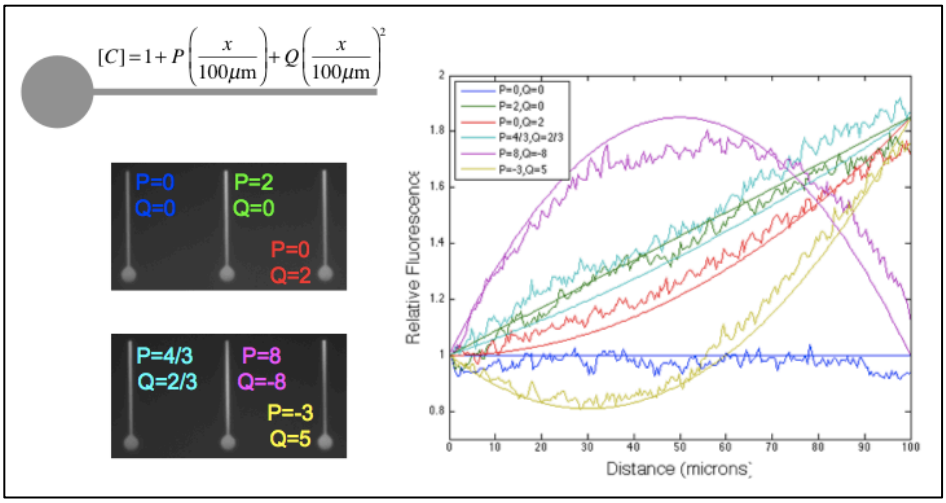

Figure 2.4. Arbitrary fluorescein gradients created using the LAMP system. The functional forms of the 100 μm-long gradients were quadratics, with specified coefficients. The graph on the right shows the measured fluorescence of each gradient as well as the expected profile.

Figure 2.5. Micron-scale replication of George Seurat's "A Sunday Afternoon on the Island of La Grand Jatte," using fluorescent streptavidin that adhered to patterned B4F.

Chapter 3

Semi-automated techniques for processing images of neurons

In this chapter, we describe methods we developed to accelerate image processing, which here consisted of several components: improving the contrast of the images, identification of salient features (e.g., neurons, neurites) in the images, and finally an analysis of these features. The image processing we performed was semi-automated, as at least one of these components was automated in each of the experiments.

3.1 Processing bright field images of neurons

Manually processing hundreds or thousands of images is incredibly laborious and time-consuming. For bright field imaging, this difficulty is compounded by the fact that individual cells are not very visible, due to their thinness and transparency. Imaging modalities that enhance contrast include phase contrast microscopy, Hoffman modulation contrast, dark field microscopy, etc. However, these modalities often require additional apparatuses, and phase contrast microscopy is not compatible with multi-well plates used for high-throughput studies.

Therefore, in many of our experiments, we acquired low-contrast bright field images, and then enhanced their contrast via image processing. We found that an adaptive histogram equalization (AHE) algorithm was generally sufficient for making neurites visible (see Figure 3.1). Histogram equalization reassigns pixel brightness based on ordinality, so that the pixel brightness histogram is uniform over the entire range of values (e.g., 0 to 255). While appropriate for enhancing contrast in images with relatively narrow dynamic ranges, it does not work as well on images containing multiple regions (i.e., cells) that are significantly lighter or darker than the rest of the image. AHE applies histogram equalization multiple times over smaller neighborhoods of pixels within the entire image. In this way, darker or lighter regions of the image are similarly enhanced in contrast. The high-contrast images produced by AHE were more readily labeled for neuronal cell bodies, neurites, and neurite tips.

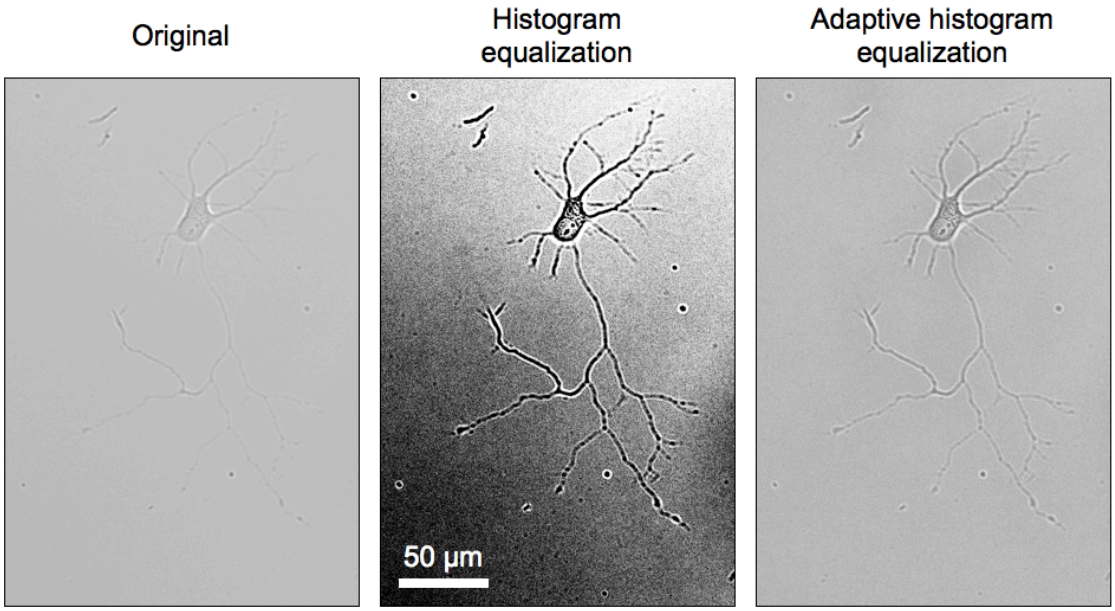

Figure 3.1. Histogram equalization algorithms for improving the contrast of images of neurons *in vitro*. In the original image, the cell body is somewhat visible, but neurites are not. After a histogram equalization is applied to the image, the entire neuron becomes more visible. However, histogram equalization also produces significant background variation, that can make some neurites difficult to see. AHE produces images in which neurons and neurites are visible and suitable for marking up.

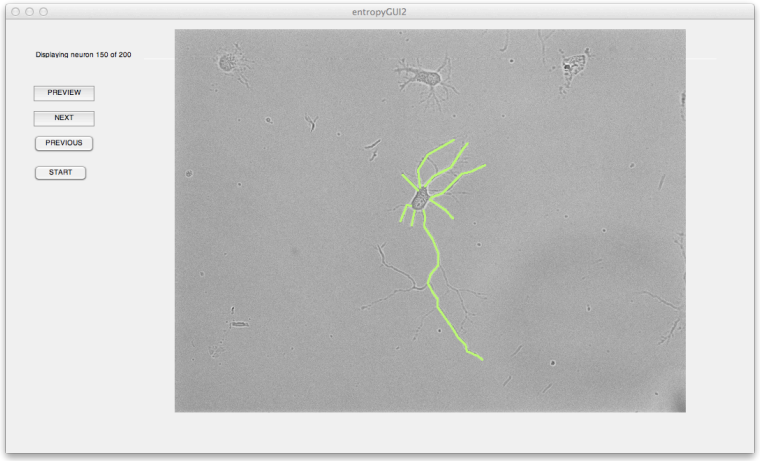

Figure 3.2. Graphical user interface for quickly marking up neurites. The image is the same as the one shown in Figure 3.1 (adaptive histogram equalization). Neurites have been labeled in green. The user can quickly scan through images of different neurons, as well as multiple time points of the same neuron.

For rapid manual processing of neuron position and neurite length, we created a custom MATLAB script and graphical user interface (GUI). For neurons that grew on linear protein substrates (Chapter 4), users only had to mouse-click on each cell body and the tips of the neuron's two neurites. For neurons that

developed freely on two-dimensional substrates (Chapter 5), users could trace each neurite with a series of left clicks, and neurite tip with a single right click (Figure 3.2). For both substrates, neurite lengths and neuronal polarity were then automatically calculated.

3.2 Processing images of neurons in microwells

The work described in this section was used in Shi, Scott, Ghosh, Wan, Wissner-Gross, et al. [35].

Figures in this section were created by Peng Shi.

In Chapter 5 we discuss cells whose positions can vary in two dimensions, while in Chapter 4 these positions are constrained to a single dimension. But when a cell is constrained to a single point, the image processing can be fully automated without sacrificing the quality of the data.

We created such a high-throughput apparatus, which we called a "synapse microarray," in which cells were constrained to very specific locations, for studying synaptogenesis, the formation of synapses between neurons. In this experiment, HEK293 cells were loaded into 30 μm diameter polydimethylsiloxane (PDMS) wells, raised 3 μm above a bottom surface so that neurites growing out from another chamber could interact with the HEK293 cells. The HEK293 cells were treated with a variety of screening factors to determine which, if any, up- or down-regulated the interaction (i.e., synapse formation) between the neurons and the HEK293 cells.

The HEK293 cells were found exclusively in the PDMS wells and their morphology was universally circular, as opposed to the various morphologies they exhibit when growing freely on a two-dimensional substrate (Figure 3.3). This uniformity allowed us to automatically detect neurons that had been labeled with Human influenza hemagglutinin-tagged neuroligin-1 (HA-NLG1) (Figure 3.3C). We then automatically quantified synapsin expression inside the HRK293 cells, which themselves had been automatically detected (Figure 3.4). Synapsin expression is an indicator for synaptogenesis, and we measured how this expression varied under the application of a library of chemical compounds. We specifically identified a histone deacetylase (HDAC) inhibitor that enhances NLG1-induced synaptogenesis by modulating class-I HDACs.

Moreover, by fully automating the image processing, we were able to accelerate the time required for image analysis by more than 30-fold vs. manual processing (Table 3.1).

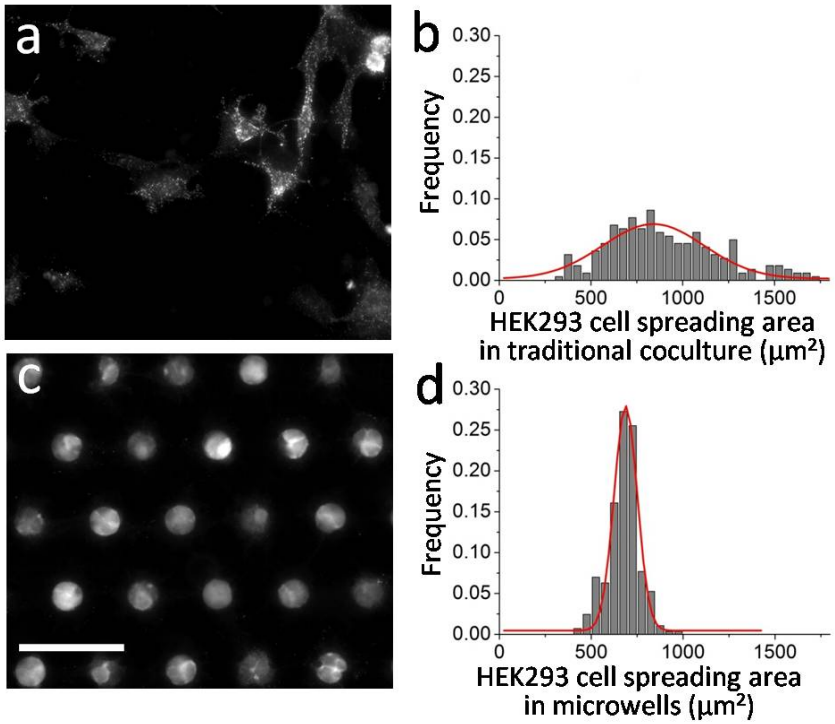

Figure 3.3. Improved uniformity of HEK293 cell contact area with neurites. (**A**) HEK293 cells spread in varying morphologies in standard cultures (staining for HA-NLG1), scale bar: 100 µm. (**B**) Histogram of HEK293 cell area in standard cultures (n = 222). (**C**) HEK293 cells grow in well-defined morphologies in the synapse microarray (staining for HA-NLG1). (**D**) Histogram of HEK293 cell area in the synapse microarray (n = 286). Curves in the histograms are fitted Gaussian distributions. Figure taken from [35].

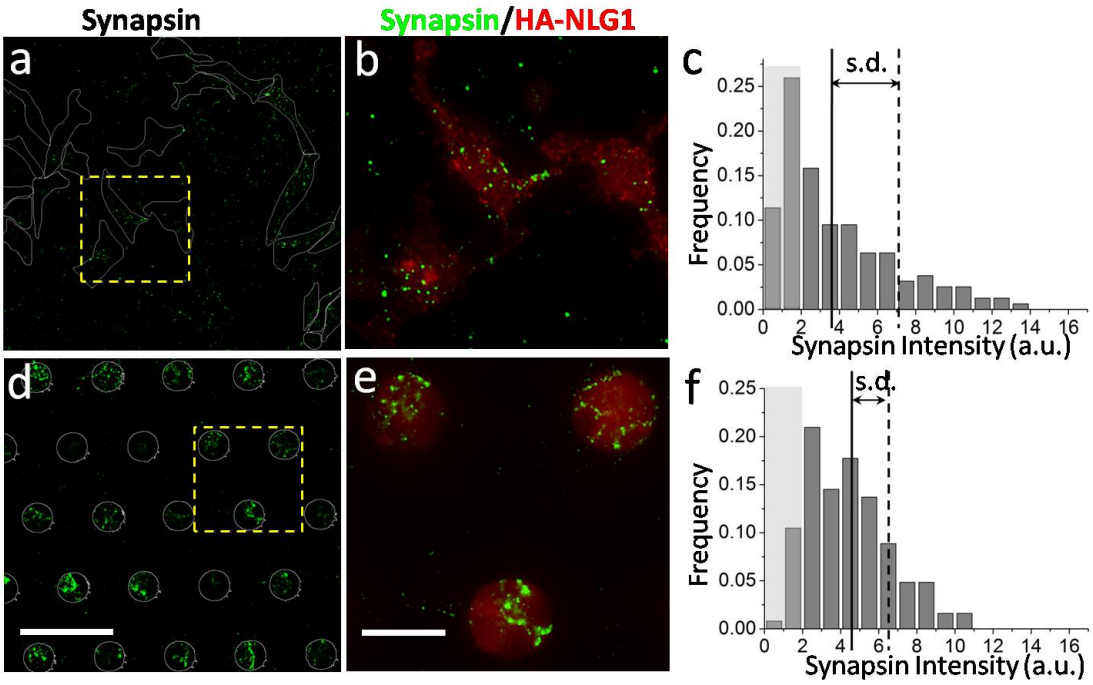

Figure 3.4. Quantification of induced synapse formation. (**A**) Fluorescence image of synapsin clustering associated with NLG1-transfected HEK293 cells (outlined in white) in standard cocultures. (**B**) Enlarged view of the boxed region in panel (**A**) (synapsin, red; HA-NLG1, green). (**C**) Histogram showing the distribution of synapsin fluorescence intensity within HEK293 cell covered regions in traditional cocultures (n = 160). (**D**) Fluorescence image of synapsin clustering in microwells (white circles) filled with HEK293 cells. (**E**) Enlarged view of the boxed region in panel (**D**) (synapsin, red; HA-NLG1, green). (**F**) Histogram showing the distribution of synapsin fluorescence intensity in microwells (*n* = 248). In panel (**C**) and (**F**), light gray covered columns indicate the portion of regions with substantially low synapsin fluorescence. One standard deviation from the mean values (solid lines) is indicated by the dashed lines. Figure taken from [35].

Table 3.1. Comparison of estimated time per sample. The synapse microarray, with automated image processing, captures and analyzes more than four times as many data points in about 10% of the time as a traditional coculture. Table taken from [35].

	Synapse microarray	Traditional coculture
Number of data points	>200	<50
Acquire 10 images	~100 s	~250 s
Outline ROI	~20 s	~1000 s
Measure and log data	~80 s	~250 s
Total time	~200 s	~1500 s
	Automated	Manual

3.3 Semi-automated measurement of neuronal migration in the cortex

The work described in this section was used in Peschansky, Burbridge, Volz, Fiondella, Wissner-Gross, et al. [36].

Neuron migration analysis is another image processing task that is often performed manually, but which could be rapidly accelerated if performed in a semi-automated manner. To allow for faster processing of neuron migration data, we created NeuronGUI (available in its entirety in Appendix B), a program that significantly speeds up image processing while simultaneously extracting much richer data from images. We specifically used NeuronGUI in a study of how embryonic knockdown and overexpression of the candidate dyslexia gene homolog Kiaa0319 affected neuronal migration in the embryonic rat cortex [36].

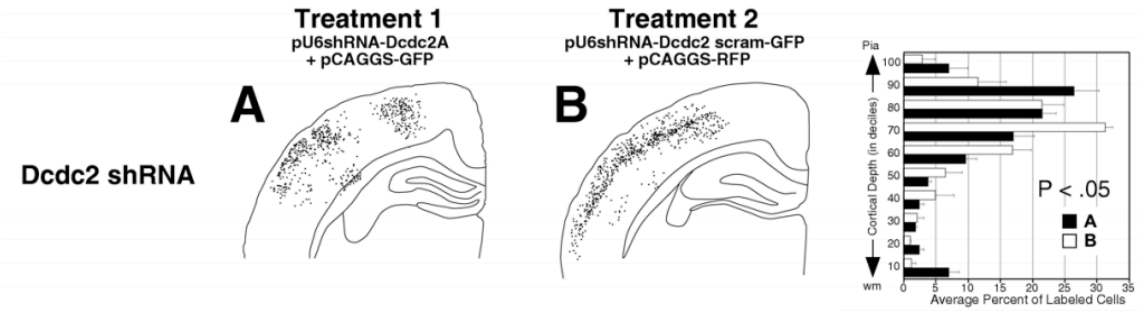

Figure 3.5. Typical manual quantitative analysis for neuronal migration. Neurons in cortex slices are labeled by hand using Neurolucida software. The relative cortical depths of the individual neurons are then manually determined, and divided into deciles. The resulting histograms (right) are compared using a χ^2 test, which is not ideal for this comparison. Figure taken from [37].

Previously, the positions of neurons in fixed brain slices were manually recorded, along with an outline of the cortex and ventricles (Figure 3.5) [37]. Neuronal migration can be quantified as the relative distance that neurons have travelled from the white matter/subplate border to the pial surface, scaled to a percentage. The distance from each neuron to either surface was manually determined, and neurons were then binned into deciles. The distributions over these deciles were then compared using a χ^2 comparison test, which is more appropriate for categorical data than the numerical data of the distributions. For example, if Experiment A resulted in all neurons migrating 91% while experiment B resulted in all neurons

migrating 89%, then a χ^2 test would say that these results are just as different as if Experiment A had all neurons migrating 100% and Experiment B had all the neurons at 0%.

NeuronGUI improves this analysis by first parsing a text file, exported from a Neurolucida file that has been manually marked up by a user, and automatically plotting an outline of the cortex along with the precise postitions of all the identified neurons (Figure 3.6). Then, rather than measure the positions of all the neurons, the user simply has to specify which vertices belong to the white matter/subplate border (which are then colored green) and which vertices belong to the pial surface (colored in red) by clicking on them.

The relative migration of every neuron is then automatically computed to an accuracy that is only limited by the user's ability to determine the coordinates of the neurons, the subplate border, and the pial surface. To determine the migration of each neuron, the distances to the subplate border (L_{sb}) and pial surfaces (L_{pia}) must first be determined, and migration is then calculated using the following formula:

$$\text{migration} = \frac{L_{sb}}{L_{sb} + L_{pia}} . \tag{3.1}$$

To determine both L_{sb} and L_{pia} for a specific neuron, NeuronGUI first finds the two connected vertices of the desired surface that are nearest to that neuron. The distance is then calculated as the minimum distance (i.e., the projection or the distance to the nearer vertex) to the line segment connected those two vertices. Thus, the two distance vectors formed by L_{sb} and L_{pia} need not be parallel (in general, they are not).

Figure 3.6. Screenshot from "NeuronGUI," an original MATLAB program we created for the rapid quantification of neuronal migration. Neurons of interest (blue) as well as relevant surfaces (white) of the brain slice are first manually identified in Neurolucida. The Neurolucida file is then exported to a text file, which is parsed by NeuronGUI. Users then click on the vertices of the white matter/subplate border (green), and of the pial surface (red). Neuronal migration is then automatically calculated for all of the neurons.

The output of NeuronGUI applied to several brain slices can be found in Figure 3.7 [36]. Rather than a histogram with deciles, neuron migration is displayed as a continuous function of cortical depth, and differences between samples are more easily detected. In this study, it was found that *Kiaa0319* short hairpin RNA (shRNA) disrupts neuronal migration by both cell autonomous and non-cell autonomous mechanisms, further linking the candidate dyslexia gene to abnormal migration patterns in brain development.

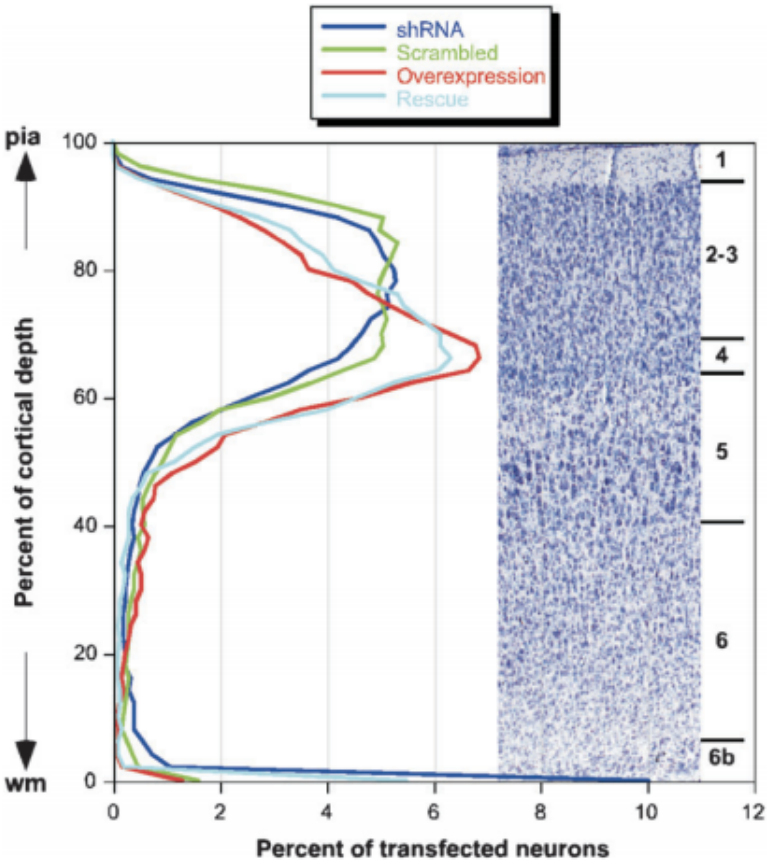

Figure 3.7. Improved quantitative analysis for neuronal migration. Slices are labeled by hand using Neurolucida software. The relative cortical depths of the individual neurons were determined using NeuronGUI, which produces the smooth migration profiles on the left. Image taken from [36].

Chapter 4

Symmetry breaking in neurons with two neurites

Sections 4.1-4.6 of this chapter are adapted from in Wissner-Gross et al. [38]. *The figures in Section 4.7 are taken from Scott, Wissner-Gross, et al.* [33]*, and were created by Mark Scott.*

During development and regeneration, neurons undergo several complex morphological and functional changes, including neurite outgrowth, axon specification, branching, and synaptogenesis. Subtle abnormalities in these processes have been implicated in several neurological disorders such as autism [39,40], schizophrenia [41], and epilepsy [42]. Developing assays for repeatable recapitulation of these complex neuronal behaviors on a large scale could allow high-content screens and therapeutic discoveries. However, due to the significant complexity and heterogeneity of neuronal dynamics, the development of such assays has proven challenging.

The dynamics of immature neurite growth and subsequent axon specification during development have been most extensively studied in primary hippocampal neurons from embryonic rats. Developing hippocampal neurons extend several immature neurites, one of which ultimately becomes the longer axon while the remaining neurites become the shorter dendrites [3,4], i.e., the neurons polarize. Axotomies of hippocampal neurons in culture have revealed that once one neurite grows 10-15 μm longer than the others, it usually becomes the axon [5,6]. The process of axon specification is believed to rely on feedback loops among the neurites, in which longer neurites promote their own growth and inhibit the growth of other neurites [4,43]. Several molecular components of these feedback loops have recently been identified [9,10,44].

Previous studies of neurite growth have typically involved neurons growing freely on a two-dimensional substrate [3,9,45–49] or along intersecting stripes [22,50]. However, measurements of reproducible neurite growth dynamics on these surfaces are difficult to perform because neurons have varying numbers of immature neurites, and because these neurites can grow in a wide variety of morphologies, allowing quantification of only simple aspects of neurite growth dynamics over many neurons. Yet primary neurons are prohibitively difficult to isolate in large numbers, and they must also be

cultured at a high cell density, thereby increasing the likelihood of cell-cell interactions and reducing the number of neurites that grow without contacting other cells.

In this chapter, we circumvent these difficulties by using a simple strategy to generate highly reproducible neurite growth behavior in a high-throughput format. In recent years, numerous methods have been demonstrated that use surface patterning to guide neuron development [18–20,22,23,28,30,46,51–54]. However, surface patterning has never been previously used to perform high-content screens for elucidating neurite dynamics. Here, we used a femtosecond laser beam to pattern micron-wide lines of poly-D-lysine (PDL) onto a poly(ethylene glycol) (PEG) monolayer, as described in Chapter 2. Hippocampal neurons preferentially adhered to the PDL lines, and the narrowness of these lines prevented each neuron from growing more than one neurite in each direction. By imaging the neurite growth dynamics of only about a hundred cells, we acquired thousands of data points with very low signal-to-noise ratio. In particular, we showed that neurite growth accelerates until the point at which neurons polarize. We also showed that immature neurites compete, even at very short lengths, and neuronal polarity exhibits a distinct phase transition as neurites grow.

Such quantification of neurite dynamics puts significant constraints on possible mechanistic models of neurite growth. A number of mathematically and biophysically inspired models of neurite outgrowth have been proposed in recent years [7–9,14–17,43,55,56]. In these models, small differences in neurite lengths and growth rates are amplified over time until one neurite becomes the axon, which continues to grow at a fairly constant rate, while all the other neurites become shorter dendrites or retract [3]. However, in the absence of large-scale quantitative experimental data, these different models have remained similarly plausible.

We selected three representative models, which we compared to our experimental findings. The simplest of these models describes neurite outgrowth as a competition for resources produced by the soma at a fixed rate and distributed to the different neurites as a function of their lengths [7]. More detailed biophysical models include the growth-dependent transport and diffusion of raw materials for neurite growth between the soma and the neurite tips [8,15]. Recently, a more complex and biochemically supported model was constructed based on a positive feedback loop between the GTPase HRas and phosphatidylinositol 3-kinase (PI3K) at the neurite tip [9].

By finding the best fits of these models to our neurite growth data (i.e., neurite length as a function of time), we first determined the unknown parameters used in these models. We then examined how well these models predicted neurite growth and competition dynamics. None of these models was fully consistent with our measurements. We found that although specific modifications of the complex molecular model of Fivaz *et al*. [9] allowed this model to better recapitulate our observations, these modified versions also did not completely agree with our results. Thus, such high-content measurements can put significant and nontrivial constraints on possible mechanistic models of neurite growth.

4.1 Models of developing neurons with two neurites

Each neurite growth model we studied consists of several equations as well as parameters such as rate constants and diffusion/transport constants (Figure 4.1), which were assumed to be the same for all neurons. Neurite lengths for the Khanin [7] and Samuels [8] models were calculated as a function of time using a fourth-order Runge-Kutta-Fehlberg differential equation solver (MATLAB function *ode45*), while a multistep stiff differential equation solver (MATLAB function *ode15s*) was used to calculate neurite lengths for the Fivaz *et al*. model.

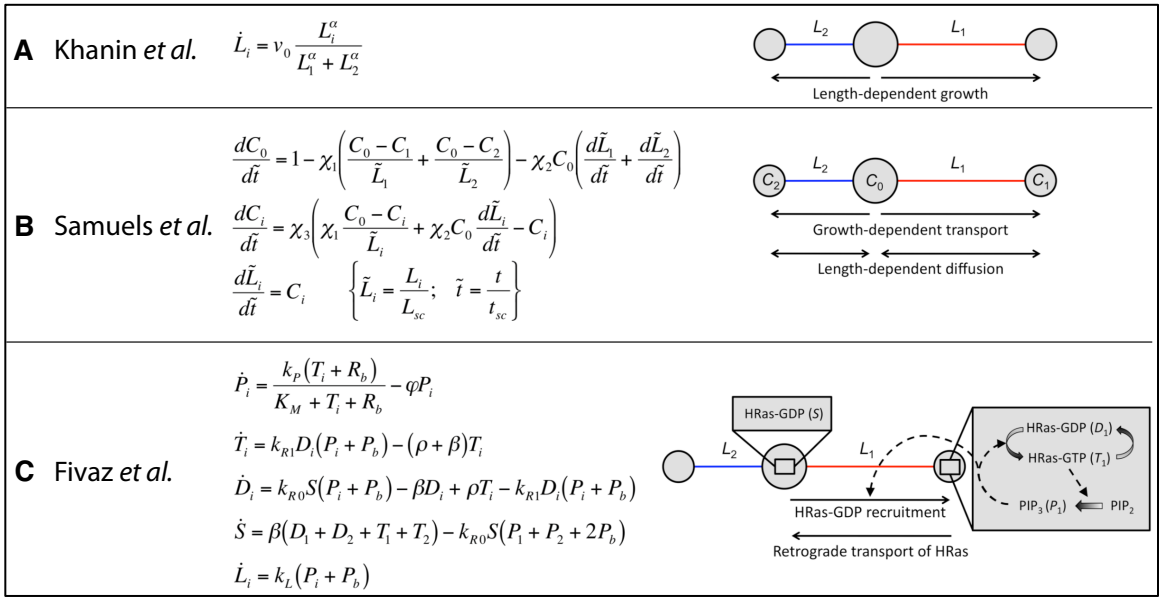

Figure 4.1. Models for the growth of two competing neurites. The subscript "*i*" is either 1 or 2, indicating the neurite. The larger and smaller circles indicate the soma versus neurite tips, respectively. (**A**) The Khanin model involves the lengths of the two neurites (L_1, L_2), but no chemical concentrations. In this model, v_0 is the characteristic growth rate and α represents the strength of the competition between neurites.

Figure 4.1 (Continued). (**B**) The Samuels *et al*. model includes the concentrations (C_0, C_1, and C_2) of an unknown factor that is rate-limiting for neurite growth. The growth of a neurite is proportional to the concentration of the factor at that neurite's tip, and the factor undergoes both diffusion and length-dependent anterograde transport. Here, L_{sc} and t_{sc} are characteristic length and time scales, respectively, while χ_1, χ_2, and χ_3 are dimensionless constants. (**C**) The Fivaz model describes the dynamics of multiple molecular species, including HRas, PI3K, PIP_2, and PIP_3. Phosphorylated HRas stimulates PI3K activation and PIP_3 production at the neurite tips. In turn, PIP_3 stimulates HRas phosphorylation in a positive feedback loop and the recruitment of additional HRas from the soma. In this model, k_P, k_{R0}, and k_{R1} are rate constants, φ and ρ are rates of protein degradation, β controls the rate of return of HRas to the soma, P_b and R_b are the respective baseline concentrations of PIP_3 and HRas, K_M is the concentration of HRas for half-maximal production of PIP_3, and k_L is the characteristic neurite growth rate. The Fivaz model also incorporates stochastic vesicular transport, as discussed in the supporting information.

4.2 Patterning stripes of poly-D-lysine

To create narrow protein stripes preventing growth of more than single neurites, an optical patterning technique was used with sub-micron resolution. Poly-D-lysine patterns were created using an improved version of laser-assisted protein adsorption by photobleaching (LAPAP) [31,32]. The original LAPAP technique does not allow reliable growth of neurites along well-defined protein micropatterns, since cell bodies and neurites also grew on parts of the substrate that are not patterned. In order to overcome this challenge, rather than patterning onto a glass substrate coated with bovine serum albumin [31], we developed a new method to pattern proteins onto a glass coated with a PEG monolayer (Figure 4.2B). This significantly reduced non-specific cell adhesion and neurite growth, thereby allowing single neurites to grow on well-defined tracks and form highly repeatable morphological patterns. The glass was first cleaned with Nano-Strip (Cyantek) for 15 minutes, then rinsed in deionized water, air-dried, and placed in a toluene bath (Chromasolv Plus, Sigma). Subsequently, the bath was moved into a nitrogen bag, and PEG silane (2-[methoxy(polyethyleneoxy)propyl]trichlorosilane, Gelest) was added at a volumetric ratio of 2%. After 2 hours, the bath was removed and the glass was rinsed in deionized water, air-dried, and placed in a vacuum oven for 2 hours. Following removal from the vacuum oven, the glass was firmly attached to a silicone gasket of a 16-well plate (ProPlate slide module, Grace Bio-Labs).

We used a Ti:sapphire laser (Spectra-Physics) operating at 780 nm, near the two-photon absorption peak for fluorescein [34], to pattern fluorescein dye (100 μg/ml, J. T. Baker) onto PEG-coated glass. Laser patterning was performed using a Nikon Eclipse Ti microscope equipped with a real-time focus correction system and a 40× objective lens (NA 0.9, Nikon). The laser power at the bottom surface of

the glass slide was measured to be 7 mW, as estimated by multiplying the power level at the back focal-plane of the objective lens with its optical transmissivity. Patterns were created by scanning the laser beam at a speed of 200 µm/s using galvo mirrors (Cambridge Technology, UK) with a spatial resolution of 250 nm.

The PDL patterns consisted of 1 µm wide and 2 mm long lines with a line-to-line separation of 100 µm (Figure 4.2A). After fluorescein patterning, each well of the 16-well plate was washed with phosphate buffered saline (PBS, Sigma) and then incubated for 1 minute with PDL (10 µg/ml, 1-4 kDa, Sigma) fluorescently tagged with DyLight 549 (Pierce) to facilitate visualization. The width of the lines was confirmed by fluorescent imaging. Variation in PDL concentration along individual lines was approximately 3%. Variation in average fluorescence between lines in the same well was also about 3%. Wells were washed again with PBS before cell plating.

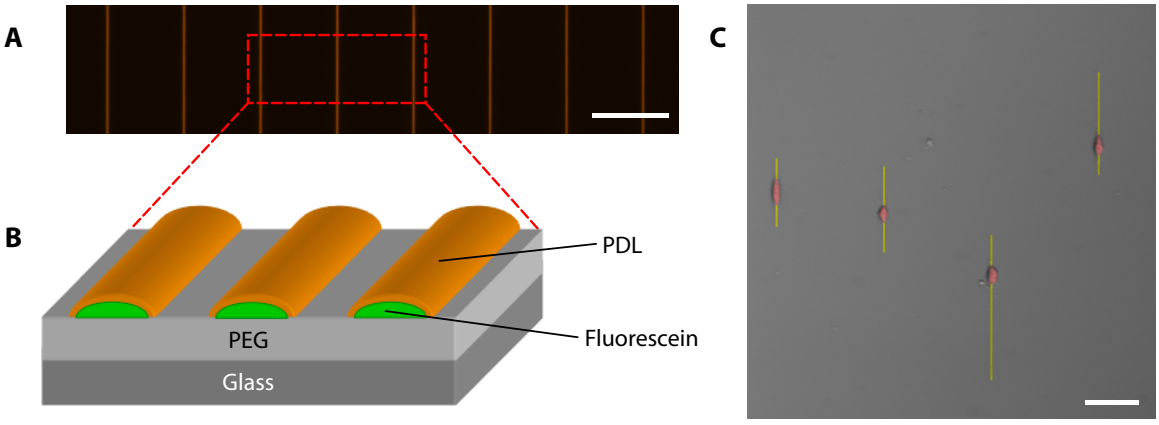

Figure 4.2. Hippocampal cell culture on PDL lines. (**A**) Fluorescent image showing long uniform lines of polylysine, which were used as a substrate for the growth of hippocampal neurons (scale bar: 100 µm). (**B**) The lines were created by patterning fluorescein onto a PEG monolayer using a femtosecond laser and then incubating in a solution containing PDL, which selectively bound to the fluorescein. (**C**) False-colored phase-contrast image showing that hippocampal neurons adhered exclusively to the PDL patterns spaced 100 µm apart. The geometry of the lines prevented the neurons from growing more than one neurite in each direction. The neurons were false-colored to enhance contrast (soma, red; neurites, yellow), but image brightness was conserved (scale bar: 50 µm).

4.3 Neuron isolation and plating

4.3.1 Isolation of rat hippocampal neurons

Hippocampi were dissected from embryonic day 18 Sprague-Dawley rats and placed in ice-cold Hank's balanced salt solution (HBSS), buffered with 10 mM HEPES, pH 7.3. The tissue was transferred to an enzyme solution containing 100 units of papain in 5 ml of HBSS, 1 mM $CaCl_2$, and 1 mM L-cystine, and subsequently incubated at 37°C for 45 minutes. The hippocampi were then washed in Dulbecco's modified Eagle's medium containing 10% fetal bovine serum to deactivate the papain, and the media was replaced with Neurobasal-B27 (Invitrogen) containing 2 mM glutamine and 100 units/ml penicillin/streptomycin. The cells were triturated using a flame-polished pipette and counted.

4.3.2 Plating neurons and verifying that single neurites grew along PDL patterns

200 ml of Neurobasal-B27 were added to each well containing the PDL patterns. Cells were then pipetted into the wells at a density of 5000 cells per well. The cells were maintained in an incubator at 37°C with 5% CO_2 and 100% humidity, and were briefly removed every hour for imaging (a sample image, false-colored to highlight cell bodies and neurites, is shown in Figure 4.2C).

Immunohistochemical stains of tubulin confirmed that single neurites, rather than neurite bundles, grew out from the neurons along the PDL lines (Figure 4.3).

Cells plated on PDL patterns for 12 hours were fixed for 15 minutes in 4% paraformaldehyde and then rinsed with a solution of phosphate buffered saline with 0.05% Tween 20 (PBST). Next, the cells were permeabilized with 0.1% Triton X-100 for 10 minutes, and washed again with PBST.

Prior to antibody staining, the surface was blocked with 3% bovine serum albumin for 30 minutes. Cells were incubated in a 1:200 dilution of mouse monoclonal anti-tubulin (primary antibody, R&D Systems) for 1 hour, and then in a 1:500 dilution of Alexa Fluor 488 goat anti-mouse (secondary antibody, Invitrogen) for another hour, with PBST washing between steps. Finally, the cells were incubated in a 1:10000 dilution of DAPI for nuclear staining, washed again, and fluorescently imaged.

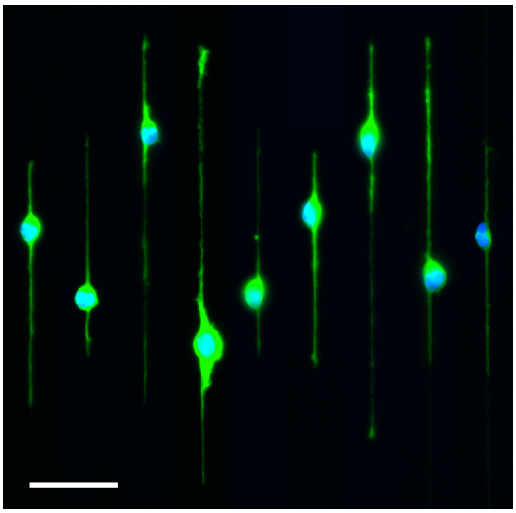

Figure 4.3. Immunohistochemical staining of hippocampal neurons and their neurites. Nuclei are shown in blue (DAPI) and tubulin in green. The average full-width at half-maximum of the tubulin profiles was 1.15 μm, and showed little variation (± 0.15 μm, SD), consistent with the diameter of single immature neurites of hippocampal neurons on two-dimensional PDL-coated surfaces (scale bar: 50 μm).

4.4 Fitting the models to neurite growth data

Rat hippocampal cells were isolated and plated as described in Section 4.3. Neurons were imaged hourly for 18 hours after plating using a CoolSnap HQ2 CCD camera. The motorized microscope stage (Prior ProScan) automatically scanned the sample to capture the entire patterned region.

We made a total of 3872 neurite length measurements, where individual neurons were imaged at 1-hour intervals for 18 hours. A representative measurement of a neuron is shown in Figure 4.4A. On average, neurite growth in both directions along PDL lines was symmetric, with mean values of 33.2 ± 0.6 μm and 32.9 ± 0.5 μm (SEM, Figure 4.4C), indicating that neurite outgrowth was not biased due to any anisotropy of the PDL patterns.

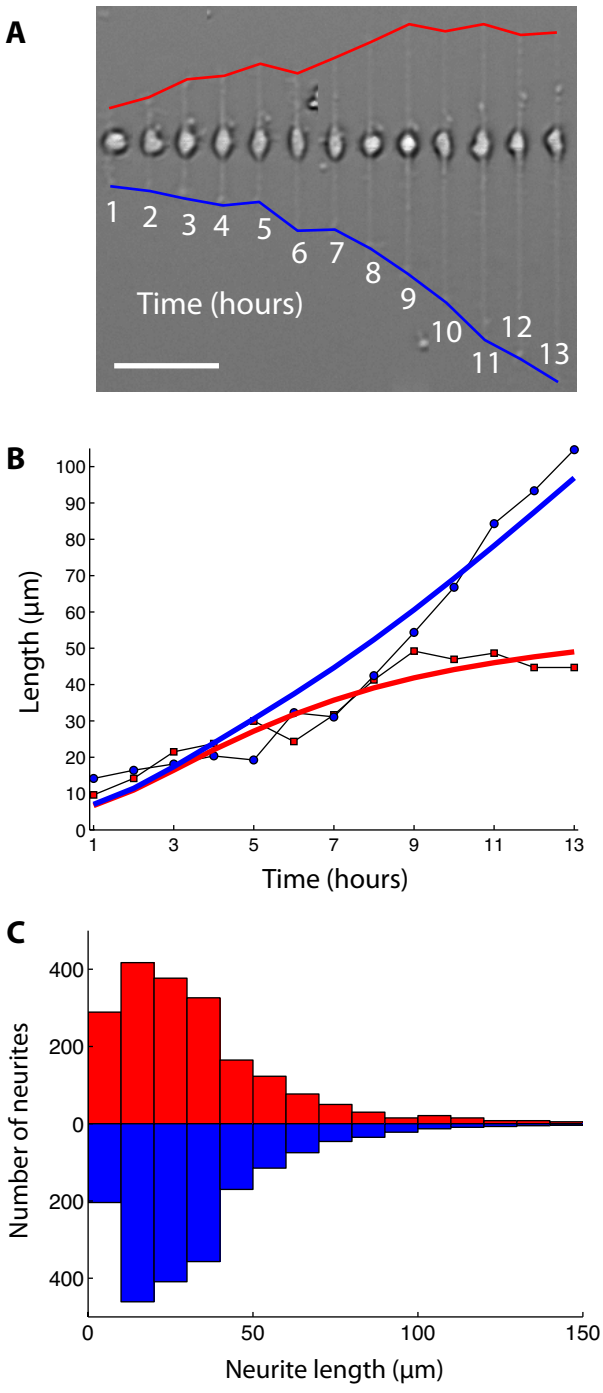

Figure 4.4. Neurite growth. (**A**) Time-lapse image of a neuron. The blue and red lines mark the extent of neurite growth over the course of 13 hours (scale bar: 50 µm). (**B**) The red squares and blue circles mark the measured neurite lengths for the same neuron. The dashed lines represent a best fit of the Samuels *et al*. model to the experimental data (t_{sc} = 4.3 h, L_{sc} = 46 µm, χ_1 = 3.2, χ_2 = 39, and χ_3 = 9.2). (**C**) Histogram of neurite lengths in one direction (red) versus the opposite direction (blue), where neurite lengths at all time points (i.e., from 1 to 18 hours) are included.

Best-fit values for the unknown parameters in the Khanin and Samuels models as well as initial conditions in all three models were determined using a gradient descent optimization scheme (in which different parameters were optimized using different gains), with additional constraints placed on the parameters as necessary, such as restricting values of physical quantities to be positive. Uncertainties in the estimation of these parameters were determined by randomly varying the initial conditions and the initial parameter values used in the optimization.

Importantly, the best-fit values of all unknown parameters were varied to assure that the general neurite growth behaviors (i.e., neurite growth rate, competition, polarization) that we study in the following sections are not affected by the choice of these parameters.

The simplest model for neurite growth, proposed by Khanin *et al*. (Figure 4.1A) predicts the specification of a single axon among several competing neurites [7]. The model assumes that the summation of the growth rate of all the neurites remains constant over time, and that the growth rate of an individual neurite increases with the neurite's length. We determined the two parameters of this model to be v_0 = 7.2 ± 0.4 μm/h and α = 1.22 ± 0.08 (SEM), where v_0 is the characteristic growth rate, and the dimensionless α represents the strength of the competition between the neurites (Figure 4.1A). This result falls within the regime where axon specification occurs ($\alpha > 1$) [7].

Samuels *et al*. proposed a model of intermediate complexity (Figure 4.1B), consisting of five coupled differential equations in the case of two competing neurites [8]. A sample fitting of this model to neurite data is shown in Figure 4.4B. The model was based on the diffusive and active transport of a rate-limiting factor for neurite growth, although the identity of this factor was unknown. Similar models have proposed that this rate-limiting material is tubulin [14,15]. The Samuels model consists of six parameters: three dimensionless constants (χ_1, χ_2, and χ_3), and a characteristic length (L_{sc}), time (t_{sc}), and a characteristic concentration (C_{sc}). We reduced the number of these parameters to five by normalizing concentrations C_0, C_1, and C_2 with respect to C_{sc}. The measured values for the parameters were t_{sc} = 6.4 ± 0.5 h, L_{sc} = 51.3 ± 4.6 μm, χ_1 = 5.7 ± 0.9, χ_2 = 36.8 ± 3.8, and χ_3 = 5.2 ± 1.3. These values for χ_1 and χ_2 fall within the regime where axon specification occurs [8].

The third model we compared to our data was proposed by Fivaz *et al*. (Fig 4.1C), who identified a positive feedback loop between HRas and PI3K at the neurite tip as the primary recruiter of additional

HRas to the neurite tip and as a driving mechanism for neurite growth and competition [9]. We used similar values for all parameters to those suggested by the authors, as these already produced good fits to the measured neurite lengths as a function of time [9].

4.5 Phenotypic behaviors of developing neurons with two neurites

To quantify the dynamics of neurite growth behavior, we first analyzed *total neurite growth rate* to see whether total growth remained constant or was accelerative, since different neurite growth models make different predictions on this point. As the neurites grow, they also compete against each other. Thus, *neurite competition* was the second metric used in our analysis. Finally, we studied the overall *neuronal polarity*, which we expected to become more observable as one neurite out-competed the other.

For these quantifications, we first defined growth velocity and acceleration for each of the neurites as a function of time. For a neurite with length $L(t)$, where t is the number of hours that have elapsed since the beginning of the observation, that neurite's velocity (in units of μm/h) was calculated using the relation

$$\dot{L}(t) = \frac{L(t+1) - L(t-1)}{2}, \tag{4.1}$$

and acceleration (in units of $\mu m/h^2$) was calculated using

$$\ddot{L}(t) = L(t+1) - 2L(t) + L(t-1). \tag{4.2}$$

4.5.1 Total neurite growth rate

We observed that the total neurite growth rate increased with total neurite length, i.e., total neurite growth was accelerative (Figure 4.5A), up to a length of approximately 80 μm. This acceleration was fastest when both neurites were very short, although there was greater uncertainty associated with these measurements.

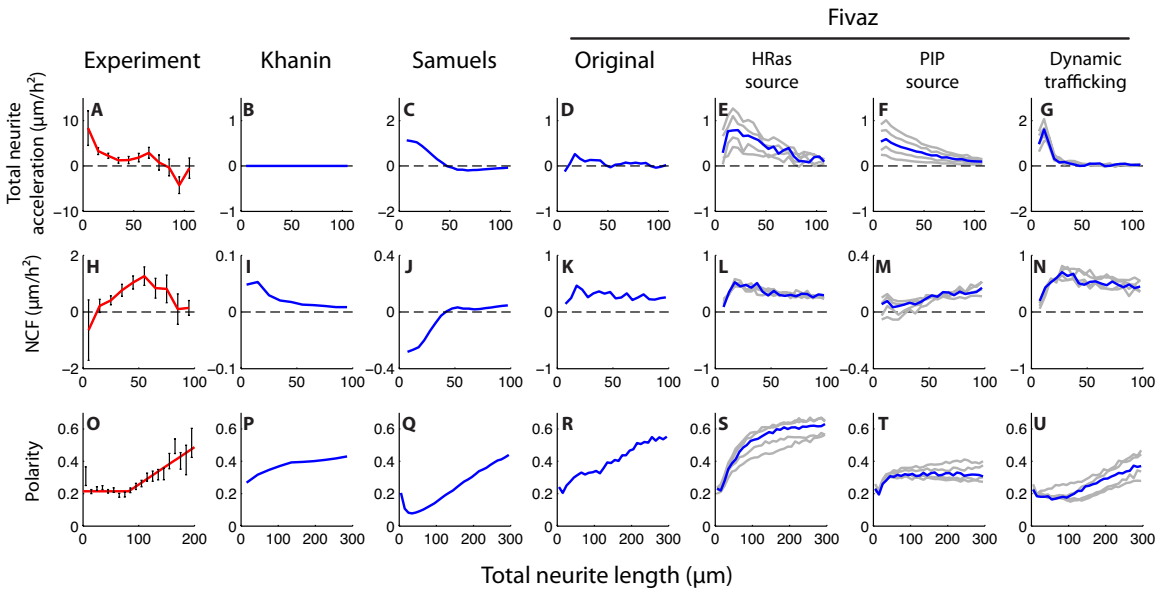

Figure 4.5: Competition between the two growing neurites and the total neurite growth rate as a function of neurite length. (**A-G**) Total neurite growth acceleration (defined as in Eq. (4.2)) vs. total neurite length (L_1+L_2). For the Khanin and Samuels models, 500 trials were averaged with randomized initial conditions. For variations of the Fivaz model, 200 trials were averaged, also with randomized initial conditions and stochastic vesicular transport. (**H-N**) Neurite competition factor (NCF), as defined in Eq. (4.3), vs. total neurite length. Again, 500 trials were averaged for the Khanin *et al*. and Samuels *et al*. models, while 200 trials were averaged for the Fivaz *et al*. models. (**O-U**) Neuronal polarity, as defined in Eq. (4.5) vs. total neurite length. Error bars in (**A**), (**H**), and (**O**) represent the SEM. Three different variations of the Fivaz *et al*. model were analyzed: "HRas source" included a constant production term for HRas in the soma; "PIP source" included a constant production term for PIP_2, which was instantaneously transported to the neurite tips and phosphorylated, becoming PIP_3; "dynamic trafficking" involved an increasing, time-dependent rate of the transport of vesicles containing HRas from the soma to the neurite tips. These variations are further detailed in Section 4.5.4. The tracings in (**E**), (**L**), and (**S**) represent HRas synthesis rates (i.e., γ in the supporting information) between 0.004 and 0.020 (blue line is 0.012). The tracings in (**F**), (**M**), and (**T**) represent PIP_2 synthesis rates (i.e., κ in Section 4.5.4) between 0.04 and 0.20 (blue line is 0.12). The tracings in (**G**), (**N**), and (**U**) represent time constants for vesicle trafficking (τ in Section 4.5.4) between 4 and 12 hours (blue line is 8 hours).

To determine whether the models predicted this accelerative growth, initial neurite lengths were randomized between 1 and 5 μm based on our experimental observations. The initial normalized concentrations (C_0, C_1, C_2) in the soma and neurite tips in the Samuels model were randomized between 0 and 0.5 i.e., within the range of values produced by parametric fitting. The initial chemical concentrations (i.e., S, P_1, P_2, T_1, T_2, D_1, D_2) in the Fivaz model were randomized between 0 and twice their respective equilibrium concentrations.

The Khanin model predicts that the total neurite growth rate is constant and independent of total neurite length, and that there is no acceleration of total neurite growth (Figure 4.5B) [7]. Thus, the Khanin model is not consistent with our total neurite growth measurements.

Meanwhile, the Samuels model, predicts that the total growth rate of neurons with short neurites rapidly increases (Figure 4.5C). This prediction is consistent with our experimental results.

For the Fivaz model, the acceleration of total neurite growth does not agree with our experimental data (Figure 4.3D). Indeed, a sustained increase in growth rate is not possible in the Fivaz model because the total amount of the growth-inducing factor HRas within each neuron is fixed by the initial conditions.

We then made several simple modifications to the Fivaz model, since this model has a well supported molecular basis [9]. These modifications were made independently of each other and were each capable of recapitulating accelerative total neurite growth (although their effects on other characteristics of neurite growth dynamics, which we discuss next, were significantly different). The first modification introduced a nuclear source term for HRas that declines as neuron matures (Figure 4.5E), since the total neurite growth rate depends on the total amount of available HRas in the neuron. A second modification introduced a nuclear source term for PIP_2, which could then be phosphorylated into PIP_3 by PI3K at the neurite tip (Figure 4.5F), as the neurite growth rates are proportional to the concentrations of PIP_3 in the neurite tips. The third modification increased the rate of vesicle trafficking of HRas as the neurons developed ("dynamic trafficking" in Figure 4.5G), since increasing the rate of HRas trafficking should also increase the total neurite growth rate. These modifications to the Fivaz model are further detailed in Section 4.5.4.

4.5.2 Neurite competition

Neurite growth rates depend on the concentrations of growth-limiting factors in the neurite tips [8,9,15]. Neurites compete for these factors to increase their length [4], and neurite growth rate, as defined by Eq. (4.1), reflects the concentrations of these factors. To quantify the dynamics of this competition (i.e., changes in concentrations of growth-limiting factors), we therefore evaluated correlations in the changes in neurite growth rates (i.e., acceleration). We define the neurite competition factor (NCF) between two neurites as the normalized product of their accelerations:

$$-\frac{\ddot{L}_1\ddot{L}_2}{\left|\ddot{L}_1\right|+\left|\ddot{L}_2\right|}. \tag{4.3}$$

A positive NCF corresponds to competitive behavior between the neurites, while a negative NCF corresponds to cooperative behavior, as shown in Figure 4.6.

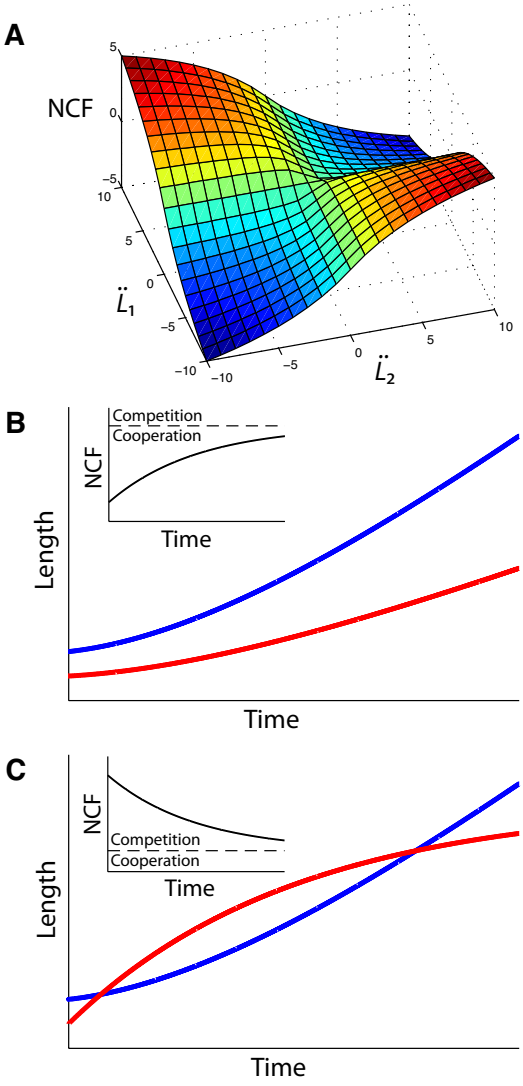

Figure 4.6. Illustration of neurite competition and cooperation in growing neurite pairs. Neurite lengths are plotted as a function of time. (**A**) Plot of the accelerations in the growth rates of two competing neurites versus the neurite competition factor (NCF), as defined in Eq. (4.3). Red indicates strong competition (positive NCF), while blue indicates strong cooperation (negative NCF). (**B,C**) Simulated examples of cooperating and competing neurites. (**Insets**) NCF as a function of time. The dashed lines indicate zero competition/cooperation. (**B**) Both neurites (red and blue) have increasing growth rates, yielding a negative (NCF), i.e., cooperation. (**C**) While one neurite's growth rate is increasing (red), the other neurite's growth rate is decreasing (blue), yielding a positive NCF, i.e., competition. The NCF in both (**B**) and (**C**) approaches zero over time as the neurite growth rates approach constant values.

We experimentally observed that as one neurite's growth rate increased, the growth rate of the other neurite on average always decreased (i.e., positive NCF in Figure 4.5H for all neurite lengths). The relative growth rates of neurites often alternated, such that both neurites grew to substantial lengths. We observed this competitive behavior over a range of neurite lengths, even when the total neurite lengths were as short as 20 μm and as long as 100 μm. This competition was strongest when the total neurite length was between 50 and 60 μm.

We next evaluated how well the different models predicted the observed competition between growing neurites. The Khanin *et al*. model predicts that the growth rate of one of the two neurites will be always increasing while the other is always decreasing. As a result, the model incorrectly predicts that the competition between neurites starts highest and monotonically decreases as neurites grow (Figure 4.5I). This prediction is inherent to the Khanin *et al*. model, and does not depend on the choice of parameters v_0 and α.

The Samuels *et al*. model also diverges significantly from the experimentally observed competitive behavior for shorter total neurite lengths (<50 μm) (Figure 4.5J). The model inherently and incorrectly predicts that the growth rates of both neurites increase (negative NCF in Figure 4.5J) until

$$\frac{L_i}{L_{sc}} > \frac{\chi_1 (C_i - C_0)}{C_i (\chi_2 C_0 - 1)} \sim \frac{2\chi_1}{\chi_2}, \tag{4.4}$$

where L_i is the length of the shorter neurite, and C_i is the corresponding resource concentration at that neurite's tip. The factor of 2 in the above relation was determined empirically from typical concentrations produced by our fitting analysis. The numerator and denominator on the right-hand side of Eq. (4.4) are proportional to the diffusion versus active transport rate of the resource. When L_i/L_{sc} is larger than the ratio of the two rates (i.e., diffusion is slower than active transport), the growth rate of the shorter neurite will begin to decrease i.e., NCF becomes positive. This transition in NCF occurs when L_i is approximately 15 μm, which implies a total neurite length of at least 30 μm, i.e., twice the length of the shorter neurite. This was in agreement with our numerical simulations of the model (Figure 4.4J). However, it was inconsistent with our experimental data in which NCF was positive even at total neurite lengths much shorter than 30 μm.

We found that the Fivaz model correctly predicted competition between the neurites, even at shorter neurite lengths (i.e., positive NCF in Figure 4.4K). Our modifications of the model also predicted competition correctly at shorter neurite lengths (Figure 4.4L-N), although the addition of a PIP source term appeared to result in an NCF that increased monotonically with neurite length (Figure 4.4M). The addition of dynamic trafficking resulted in the strongest competitive trend and a maximal NCF closest to what we observed experimentally. However, the original Fivaz *et al*. model and all modifications of the model predicted a significantly positive NCF even at longer neurite lengths, inconsistent with our experimental observations.

4.5.3 Neuronal polarization

Neurons polarize as their neurites grow, as discussed in the beginning of this chapter, and such polarization was evident in our experiments as total neurite length exceeded 80 μm (Figure 4.4O). Here, we define neuronal polarity, in the case of two neurites with lengths L_1 and L_2, as

$$P_2 = \frac{|L_1 - L_2|}{L_1 + L_2}, \tag{4.5}$$

This increase in polarity occurred quite suddenly and is suggestive of a phase transition, which occurs in various biological phenomena [57,58].

The polarization process predicted by the different neurite growth models demonstrated a variety of behaviors. For comparison, we set the initial neurite lengths in all the models to be the same to match the initial mean polarity (~0.2) we measured experimentally.

No phase transition was evident in the Khanin model, which produced a polarity that steadily increased with total neurite length (Figure 4.4P). Polarity in the Samuels model more closely resembled our experimental results (Figure 4.4Q). Like the Khanin model, the Fivaz model produced an increasing polarity (Figure 4.4R) inconsistent with our experiments.

Our modifications of the Fivaz model also resulted in distinct polarity behaviors. Introduction of an HRas source term caused the neuron to polarize at shorter lengths (Figure 4.4S), while the PIP source term appeared to lower the steady-state neuronal polarity (Figure 4.4T). These were inconsistent with the polarization behavior we measured. Our experimental observations were best predicted by the dynamic-

vesicle-trafficking modification, which resulted in a near-constant polarity of about 0.2 for total lengths under 100 µm, and an increasing polarity for lengths exceeding 100 µm (Figure 4.4U). This apparent phase transition occurs when the vesicular trafficking rate exceeds a critical value of 0.6 vesicles/minute (Figure 4.7).

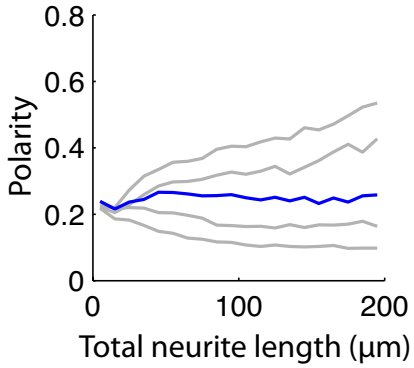

Figure 4.7. Neuronal polarization vs. vesicle trafficking rate. Each line shows how neuronal polarity evolves as a function of total neurite length for a different fixed vesicle trafficking rate. The bottom-most line was generated using a vesicle trafficking rate of 0.2 vesicles/minute, and successive lines represent increases of 0.2 vesicles/minute. The lower three rates (0.2, 0.4, and 0.6 vesicles/minute) flattened out once total neurite length exceeded 100 µm, whereas the top two lines (0.8 and 1.0 vesicles/minute) continued to polarize after 100 µm. The blue line (0.6 vesicles/minute) indicates where this behavioral transition occurs.

4.5.4 Modifying the Fivaz model

The original Fivas model consisted of the equations shown in Figure 4.1C, which are rewritten here with the stochastic anterograde vesicular trafficking terms $V_i(t)$ included:

$$\begin{aligned}
\dot{P}_i &= \frac{k_P\left(T_i + R_b\right)}{K_M + T_i + R_b} - \varphi P_i \\
\dot{T}_i &= k_{R1} D_i \left(P_i + P_b\right) - \left(\rho + \beta\right) T_i \\
\dot{D}_i &= k_{R0} S\left(P_i V_i\left(t\right) + P_b\right) - \beta D_i + \rho T_i - k_{R1} D_i \left(P_i + P_b\right) \\
\dot{S} &= \beta\left(D_1 + D_2 + T_1 + T_2\right) - k_{R0} S\left(P_1 V_1\left(t\right) + P_2 V_2\left(t\right) + 2P_b\right) \\
\dot{L}_i &= k_L\left(P_i + P_b\right).
\end{aligned} \tag{4.6}$$

Fivaz *et al*. defined $V_i(t)$ as follows:

$$V_i(t) = \frac{1}{\sigma\sqrt{2\pi}} \sum_f \exp\left(-\frac{\left(t - t_f\right)}{2\sigma^2}\right), \tag{4.7}$$

where vesicles containing HRas release their contents at the neurite tip at times t_f and σ is the temporal width of the HRas release by the vesicles. Like Fivaz *et al*., here we assume an average vesicle trafficking rate of 1 vesicle per minute for each neurite.

Our first modification of the model was to include a term representing the nuclear synthesis of HRas that declines as the neuron matures. This was accomplished by adding a decaying source term to the equation for *S*:

$$\dot{S} = \gamma \exp\left(-\frac{L_1 + L_2}{\lambda}\right) + \beta\left(D_1 + D_2 + T_1 + T_2\right) - k_{R0} S\left(P_1 V_1(t) + P_2 V_2(t) + 2P_b\right), \tag{4.8}$$

where the characteristic length λ was set to 50 μm. The magnitude γ of the source term was varied in Figure 4.5.

The second modification was the introduction of a PIP_2 source term in neurites that also declines as the neuron matures. We define Q_i as the concentration of PIP_2 in neurite i. The equation for P is then

$$\dot{P}_i = \frac{k_P\left(T_i + R_b\right)}{K_M + T_i + R_b} Q_i - \varphi P_i, \tag{4.9}$$

and we introduce a similar equation for Q:

$$\dot{Q} = \kappa \exp\left(-\frac{L_1 + L_2}{\lambda}\right) + \varphi P_i - \frac{k_P\left(T_i + R_b\right)}{K_M + T_i + R_b} Q_i, \tag{4.10}$$

Again, λ was set to 50 μm, and the magnitude κ of the source term was varied in Figure 4.5.

The third modification ("dynamic trafficking") added time dependence to the vesicle trafficking rate. This rate was initially set to zero, and exponentially approached a steady-state value of 1 vesicle per minute per neurite with a characteristic time τ, which was varied in Figure 4.5.

4.6 Protein patterning for high-content studies of neuronal development

Large-scale studies of neurite dynamics have remained difficult to quantify because neurites branch and grow in various morphologies. By growing neurons on protein micropatterns, we increased the repeatability of neurite growth behavior. This enabled us to measure and analyze the dynamics of neurite

growth acceleration, neurite competition, and neuronal polarity with unprecedented reproducibility using a minimal number of primary neurons.

We specifically found that neurites compete with each other, even when total neurite length is as short as 20 μm. This is a non-obvious behavior, as certain models predict incorrectly that both neurites can accelerate simultaneously. We also found total neurite growth is also accelerating at these shorter lengths. Then, as total neurite length increases to a critical length (in this case, about 80 μm), several events happen simultaneously: the neuron polarizes (Figure 4.5O), total neurite acceleration approaches zero as the total neurite growth rate reaches a steady-state value (Figure 4.5A), and neurite competition disappears as the axon becomes specified (Figure 4.5H). This work represents the most quantitative description of neurite growth and competition to date.

We were further able to evaluate several leading models of neurite growth by analyzing their ability to predict these fundamental biophysical behaviors of neurite growth and axon specification (Table 4.1). The Khanin *et al*. model is empirically hypothesized and incapable of reproducing behaviors similar to our experimental observations, since it presumes a constant total growth rate. The model therefore exhibits zero total neurite acceleration (Figure 4.5B). In addition, the model incorrectly predicts that the strongest competition (i.e., the highest NCF) occurs while neurites are shortest and that neuronal polarity smoothly increases with total neurite length (Figure 4.5P).

Unlike the Khanin *et al*. model, the Samuels *et al*. model proposes a chemical mechanism for neurite growth. In this model, growth depends on the concentration of a specific chemical that is actively transported from the soma to the neurite tips, that passively diffuses between these locations, and that is consumed during neurite growth. At shorter lengths, diffusion between neurites occurs rapidly, allowing the neurites to effectively share a common pool of resources when they are short. As a result, short neurites grow cooperatively until reaching lengths at which active transport begins to dominate (see Figure 4.5Q and Eq. (4.4)). Thus, while this model correctly predicts accelerative total neurite growth (Figure 4.5C), it results in cooperative rather than competitive behavior for neurites shorter than the length at which the active transport rate exceeds diffusion.

The Fivaz model includes the transport of several chemicals known to be involved in the establishment of neuronal polarity, allowing it to exhibit more complex behaviors than those of the

Samuels *et al*. model. While the Fivaz model correctly predicts that both shorter and intermediate neurites compete, the model does not predict accelerative growth at short neurite lengths (Figure 4.5D), since the total amount of HRas (necessary for neurite growth) is fixed in the model. By including source terms for HRas and PIP, we were able to recapitulate accelerative growth in the Fivaz model (Figure 4.5E-F). However, neither these modifications nor the original model predicts the sharp phase transition that occurs during neuronal polarization (Figure 4.5R-T), suggesting that some additional mechanism is required to better reproduce our experimental findings. We hypothesized that a stronger nonlinearity between neurite length and neurite growth is necessary to induce such a sharp transition in polarity. We introduced such a nonlinear mechanism in the form of time-dependent vesicle trafficking because neuronal polarization strongly depends on vesicle trafficking rate (Figure 4.7). This modification recapitulated the observed phase transition in polarity. We note that other nonlinear mechanisms are plausible that could recapitulate such a sharp phase transition in polarity, and further biochemical assays are necessary to identify the nature of this nonlinear mechanism. Although our modifications significantly improved agreement of the Fivaz model with our measurements, there were still minor but measurable differences (i.e., none of the modifications recapitulated the weaker competition between long neurites). Thus, our high-content measurements put significant and nontrivial constraints on possible mechanistic models of neurite growth that cannot be reconciled by simple modifications. While biochemical assays in the future are necessary to validate any model, quantification of neurite dynamics by our technique allows discrimination of different mechanistic models.

It is possible that restricting neuronal adhesion and neurite growth to one dimension affects neurite competition and axon specification. However, we still observe all of the neurite growth and competition dynamics previously measured with standard two-dimensional cell culture techniques. Furthermore, the mechanistic models we evaluated allow for any number of neurites, including two, and do not depend on the relative orientations of neurites. Thus, it is likely that the highly repeatable neurite dynamics we observe here extend to more complex morphologies of developing neurons.

The protein patterning and analytical strategies developed here can be extended to analyze more complex neurite dynamics. These high-content measurement techniques can allow construction and validation of models for complex neuronal processes. They could also be used for large-scale chemical and

target-based screens on a variety of complex phenotypes for therapeutic discoveries using a minimal number of primary neurons.

Table 4.1. Predictions of different models vs. experimental outcomes. Red shaded boxes indicate predictions that were inconsistent with our experimental observations. The Khanin model does not allow accelerative neurite growth, results in neurite competition that monotonically decreases with neurite length, and does not produce a phase transition in neuronal polarity. The Samuels model predicts accelerative neurite growth and results in cooperation rather than competition between short neurites. The Fivaz model does not predict accelerative growth, weak competition between long neurites, or a phase transition in neuronal polarity. Three simple modifications of the model ("HRas source," "PIP source," and "dynamic trafficking") recapitulate accelerative growth, but do not predict weak competition between long neurites. Of these modifications, only "dynamic trafficking" results in an apparent phase transition in neuronal polarity.

	Experiment	Khanin *et al.*	Samuels *et al.*	Fivaz *et al.*	HRas source	PIP source	Dynamic trafficking
Accelerative growth of short neurites	✓	✗	✓	✗	✓	✓	✓
Weak competition between short neurites	✓	✗	✗	✓	✓	✓	✓
Strong competition between intermediate neurites	✓	✗	✗	✓	✓	✗	✓
Weak competition between long neurites	✓	✓	✓	✗	✗	✗	✗
Phase transition in neuronal polarity	✓	✗	✓	✗	✗	✗	✓

4.7 Directed polarization of developing neurons

After studying neurite growth and competition on narrow, micron-wide stripes of PDL, we next determined whether we could *direct* the polarization of developing neurons. To accomplish this, we laser-patterned large arrays of triangular-shaped patterns using the ultra-rapid laser patterning technology described in Section 2.3. Two-dimensional triangular patterns have previously been used to construct "diodes," for unidirectional neuronal signaling [59]. We believed that the geometric tapering of the triangles would allow neurites to grow longer in the "forward" direction of the triangles (see Figure 4.8), while neurites growing in the "reverse" direction would be stunted by the lack of a continuous path between triangles.

We varied the size and aspect ratio of the triangles, creating six unique triangle geometries (width × height: 10 × 10 μm, 30 × 30 μm, 60 × 60 μm, 10 × 30 μm, 50 × 30 μm, and 70 × 30 μm). Each geometry

was patterned thousands of times, and hippocampal neurons dissected from E18 Sprague-Dawley rats were plated onto these patterns. After 48 hours, the neurons were immunocytochemically stained for tubulin.

Figure 4.8A-B shows low- and high-resolution images of cells plated on these patterns. Hundreds of neurons were imaged, and for each neuron, the lengths of the longest neurites growing in the forward and reverse directions (L_{forward} and L_{reverse}) were recorded. The polarization efficacy was defined as

$$\frac{L_{\text{forward}}}{L_{\text{forward}} + L_{\text{reverse}}}, \tag{4.11}$$

All six geometries resulted in significant forward polarization of neurons ($p < 0.05$, Figure 4.8C). In four of the six geometries, the mean of the longest forward neurite was significantly greater than the mean of the longest reverse neurite.

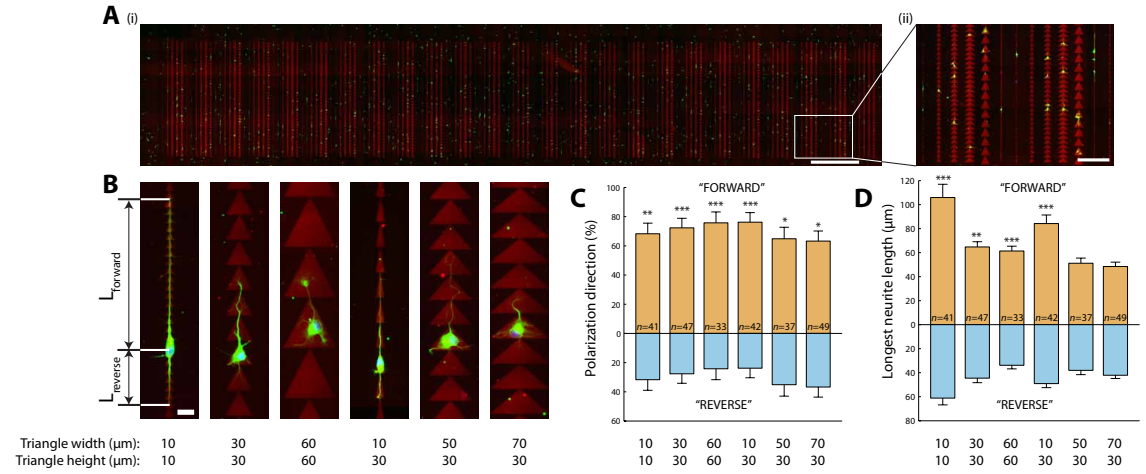

Figure 4.8. Strips of triangles of varying geometries can direct the polarization of single neurons. (**A**) A total of 15,000 patterned PDL triangles were formed over an area of 15 × 2.4 mm^2, with a ~1 μm resolution. (i) Scale bar = 1 mm, (ii) Scale bar = 200 μm. (**B**) Neurons grown on triangles with varying heights and widths. The forward direction is defined as the direction from base to apex of the triangles. Scale bar = 20 μm. (**C**) Proportion of neurons with their longest neurite in the "forward" or "reverse" direction. (**D**) Mean lengths of neurites in the "forward" or "reverse" direction. If multiple neurites were growing in one direction, the longest neurite length was used. For all graphs: * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$, comparing "forward" with "reverse" lengths.

To ensure that the neurons were polarizing on these patterns, another batch of E18 hippocampal neurons were cultured on 10 × 10 μm and 20 × 20 μm triangles for 48 hours and then stained for the axonal marker tau1 and the dendritic marker MAP2 (Figure 4.9). We found that neurons did indeed polarize on these patterns, as seen in Figure 4.9C-D.

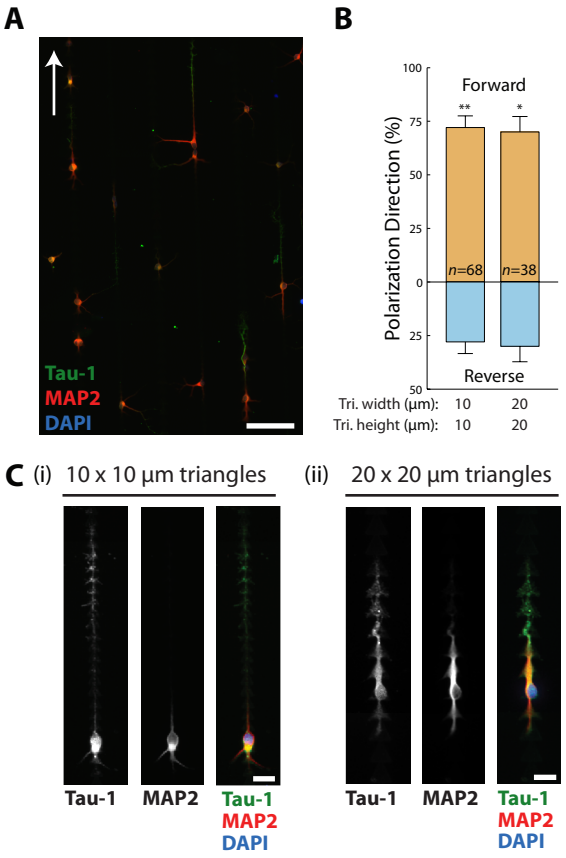

Figure 4.9. MAP2 and dephospho Tau-1 stains confirm directed polarization of single neurons. (**A**) Tau-1 labeled axons (green) preferentially grow in the direction of the triangles (indicated by white arrow). Scale bar = 100 μm. (**B**) Distribution of axons growing in the forward or reverse directions for 10 × 10 μm and 20 × 20 μm triangles. (**C**) Tau-1 and MAP2 stained neurons extending on (i) 10 × 10 μm triangles, and (ii) 20 × 20 μm triangles. Scale bars = 20 μm. * $p < 0.01$, ** $p < 10^{-4}$.

From the large image in Figure 4.8A, it became apparent that while the geometry of the patterns was very consistent, there was a noticeable variation in PDL density of the patterns (observed as a variation in pattern brightness in the fluorescent image) over the large patterning area. We then tested whether the density of the PDL patterns affected neurite outgrown and polarization, and these results are shown in Figure 4.10. For this experiment, we created 50 × 50 μm triangles, and found that neurons grew better on lower-density patterns. In fact, only for the lowest relative density (10% of the maximum density) was the longest forward neurite significantly longer than the longest reverse neurite ($p < 0.05$).

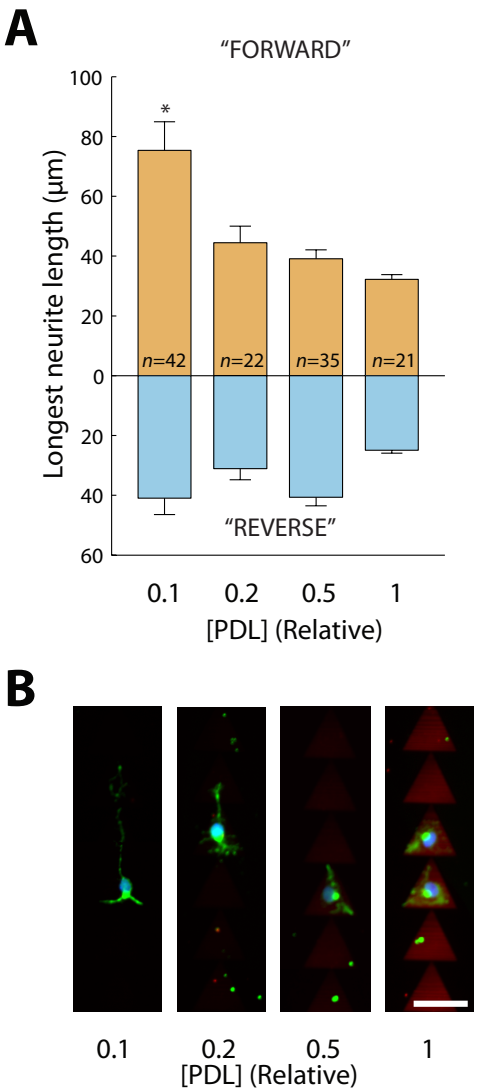

Figure 4.10: Lowering PDL density enables longer neurite outgrowth. (**A**) Mean length of neurites with varying PDL density on 50 μm triangles. (**B**) Images of neurons grown on varying densities of PDL. Scale bar = 50 μm.

Finally, we verified our hypothesis of how neurons polarize on the triangular arrays via time-lapse imaging (Figure 4.11). Figure 4.11A-B shows the behavior of a typical neuron on a 20 × 20 μm trianglular array. After a time (35 h for this particular neuron), The forward neurite grows continuously to a length exceeding 150 μm, characteristic of an axon. Meanwhile, the reverse neurite is revealed, in numerous images like Figure 4.11A(i), as being "stuck" in a "dead end," i.e., on a bottom edge of a triangle, unable to cross the PEG gap to the next triangle. After unsuccessful attempts by the reverse neurite to grow out around 22 h and again at 32 h, the forward neurite is ultimately selected as the neuron's axon.

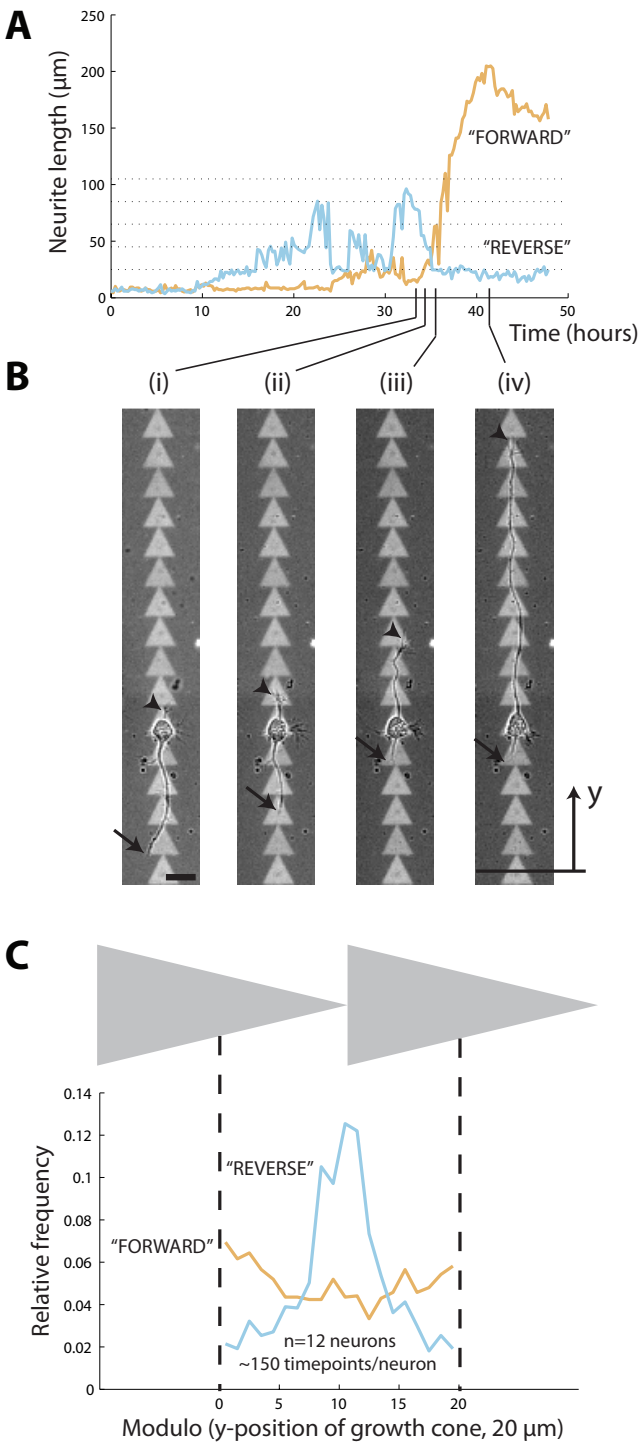

Figure 4.11. Time lapse imaging of neurite outgrowth on triangles reveals neurite polarization dynamics. (**A**) The length of forward and reverse projecting neurites from a neuron growing on 20 × 20 μm triangles (width × height) plotted over 48 h. Images were taken at 14 min intervals. The dotted lines represent the lengths at which the reverse projecting neurite would cross between two triangles. (**B**) Images of the same neuron at four time points. (i) The reverse projecting neurite (arrow) runs into a "dead end." (ii) The reverse projecting neurite retracts, while the forward projecting neurite (arrowhead) elongates.

Figure 4.11 (Continued). (iii-iv) The forward projecting neurite rapidly elongates while the reverse projecting neurite remains trapped. Scale bar = 20 μm. (**C**) A histogram derived from $n = 12$ neurons, (10 forward polarized, 2 reverse polarized), ~150 time points per neuron, showing frequency of neurite tip position as a function of its position relative to patterned triangles.

We were able to capture this behavior of multiple neurons in Figure 4.11C, which is a histogram of the vertical lengths of both the forward and reverse neurites taken modulo 20 μm, the height of a single triangle, over all the time points. If a neurite grows continuously and without interruption, then one would expect a uniform distribution in this histogram. However, if a neurite is impeded at certain locations in the triangular array, then one would expect peaks in the histograms in these locations, since the neurite will be imaged there more frequency. Indeed, we see these precise behaviors: forward-growing neurites exhibit a relatively uniform distribution, while reverse-growing neurites show a strong peak at the boundary between adjacent triangles, confirming our hypothesis.

Chapter 5

Synchronous symmetry breaking in neurons with different neurite counts

The work in this chapter is adapted from Wissner-Gross et al. [60].

In this chapter, we discuss our studies of how a neuron's neurite count affects the rate at which that neuron breaks symmetry – namely, that there is no such effect. Previously, the role of neurite count in neuronal symmetry breaking had never been examined, despite the fact that neurite count can vary significantly, and is one of the most salient properties of developing neurons. Leading biophysical models suggest that neurites compete for specific proteins (such as HRas and shootin1) in a winner-take-all fashion during axon specification [7–9,11].

These models include the Samuels model [8], the Fivaz model [9], and the Toriyama model [11]. As discussed in Chapter 1, the Samuels model was the first published model to describe axon specification using the transport and diffusion of a rate-limiting chemical for neurite growth. The Fivaz and Toriyama models are more recent, and similarly involve competition among neurites for a pool of proteins involved in neuronal polarization (HRas in the Fivaz model, shootin1 in the Toriyama model).

Since the number of interactions among competing neurites and the overall complexity of the neuron should increase with neurite count, one might expect a neuron with more neurites to polarize more slowly. Indeed, such a finding has been previously reported in these models. For example, in the Toriyama model, neurons with 10 neurites are predicted to polarize several days later than neurons with only 3 neurites. However, previous experiments have shown that neurons with varying final neurite counts all polarize within the same 48 h time window [3,61,62]. Thus, there remains a fundamental disagreement between the theoretical models and the experimentally observed biology.

In this chapter, we first verify that neurons with different neurite counts polarize synchronously (Section 5.3). We then both experimentally and computationally investigate the mechanism underlying this phenomenon, and offer simple modifications to the models so that they correctly predict rates of symmetry breaking that are independent of neurite count (Sections 5.4-5.6).

5.1 Quantifying neuronal polarity

Before comparing the polarity of different neurites, we first need a polarity metric that can be applied to neurons with different neurite counts. Here we present such a metric in Section 5.1.1, and then demonstrate its utility in observing and quantifying polarization in developing neurons in Section 5.1.2.

5.1.1 A metric for neuronal polarity in neurons with an arbitrary neurite count

Previously, the most commonly used polarity metrics have included the absolute [63–65] or relative length of a neuron's longest neurite [9]. However, these definitions have several flaws. Using the absolute length of the longest neurite discards significant information regarding the lengths of the remaining neurites. For example, if a neuron's longest neurite is 150 μm, one cannot say whether it is highly polarized: it may be polarized if the second-longest neurite is 30 μm, but not if the second-longest neurite is 120 μm. Using the relative length of the longest neurite (i.e., the ratio between the length of the longest neurite and the total neurite length) poses similar problems. By this definition, a neuron with exactly two neurites must have a polarity exceeding 0.5, while a neuron with many neurites has a wider range of polarities, making comparisons between neurons with different neurite counts difficult. Moreover, a neuron with ten neurites that are 10 μm long and a single neurite that is 100 μm long will have the same polarity as a neuron with two 20 μm neurites and one 40 μm neurite. While common sense suggests that the former neuron is more polarized, this definition produces equal polarities because it too discards significant information regarding the shorter neurites.

In Chapter 4, we introduced a quantitative metric for polarity in neurons with two neurites [38]. If these neurites have lengths L_1 and L_2, we defined the polarity, P_2, as

$$P_2 = \frac{|L_1 - L_2|}{L_1 + L_2}, \tag{5.1}$$

a metric that ranges from zero (completely unpolarized, neurites are equal in length) to unity (completely polarized, one neurite is much longer than the other). We have previously experimentally shown that this metric displays a phase transition in neurite polarization as a function of neurite length as one would expect [38].

Here, we generalized Eq. (5.1) to neurons with more than two neurites. For a neuron with N neurites of length $L_1, \ldots, L_N$, we define x_i as the normalized length of neurite i:

$$x_i = \frac{L_i}{\sum_{j=1}^{N} L_j} \,. \tag{5.2}$$

We then define the polarity using the following equation:

$$P_N = \frac{N}{2(N-1)} \sum_{i=1}^{N} \left| x_i - \frac{1}{N} \right| . \tag{5.3}$$

The normalizing coefficient $N/[2(N\text{-}1)]$ in Eq. (5.3) constrains the polarity between 0 and 1, and in the case of $N = 2$, Eq. (5.3) reduces to Eq. (5.1).

Here we have introduced a new metric for neuronal polarity that uses information about all of a neuron's neurites, unlike previously used metrics. Other metrics that meet this criterion are possible – for example, one might construct a polarity metric based on Shannon entropy, treating the relative lengths of the neurites as discrete probabilities:

$$P_{\text{Shannon}} = -\sum_{i=1}^{N} x_i \ln x_i \,, \tag{5.4}$$

However, Eq. (5.4) has a maximum of $\ln(N)$, which depends on the neurite count, making it difficult to compare polarities of neurons with different neurite counts.

The metric we used, Eq. (5.3), is relatively simple. And because of its linear nature, it is well-suited for measuring small changes in polarity when neurons are highly unpolarized. But even more important than which metric we used, we made sure to use the same metric when comparing the models' predictions with our experimental data.

5.1.2 Our metric captures polarizing behavior in developing neurons

In addition to its simplicity, Eq. (5.3) illustrates the polarizing behavior in developing neurons. Examples of polarities of neurons with different neurite counts can be found in Figure 5.1.

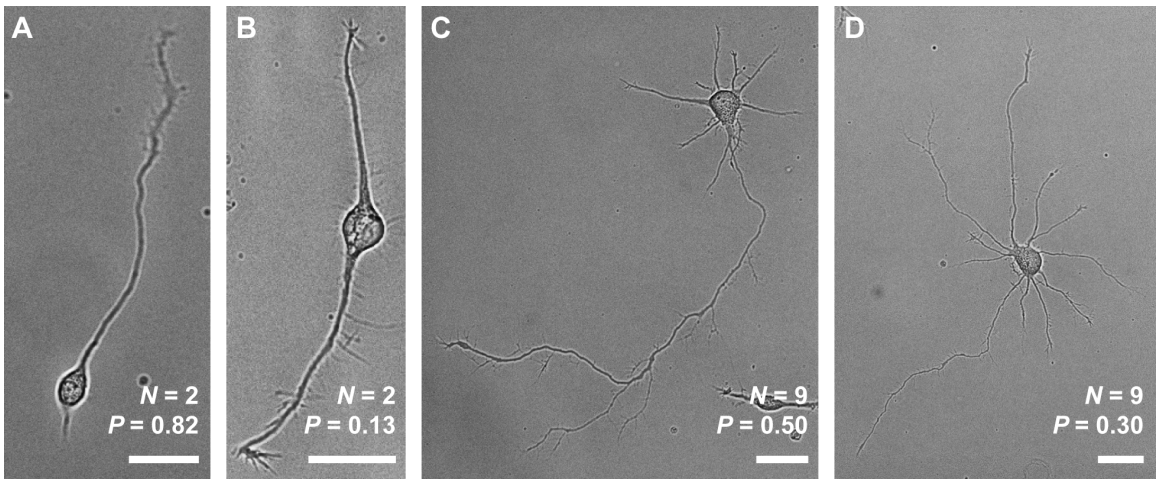

Figure 5.1. Bright-field micrographs of four different neurons. Neurite count (*N*) and polarity (*P*) using Eq. (5.3) are also indicated for each neuron. **A** and **B** show examples of neurons with two neurites that are relatively more (**A**) or less (**B**) polarized. **C** and **D** similarly show neurons with many neurites that are relatively more (**C**) or less (**D**) polarized. The image in **A** was taken 28.5 h after plating, and **B**-**D** were taken at 52.5 h after plating. All scale bars are 25 μm.

In a time-lapse study of over 130 neurons (discussed further in Section 5.3), mean polarity, as defined by Eq. (5.3), was found to increase at an approximately linear rate (Figure 5.2A, $p < 10^{-10}$). Moreover, our polarity metric captured the phase transition in polarity as total neurite length increased beyond a critical value (Figure 5.2B), consistent with our studies of neurons with exactly two neurites discussed in Section 4.4.

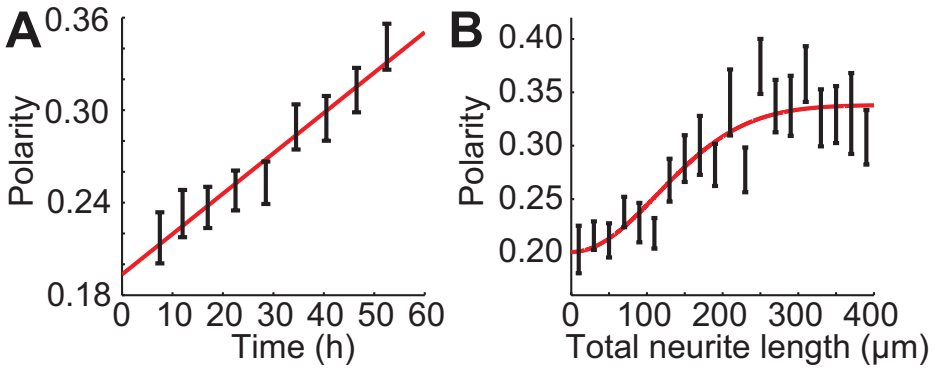

Figure 5.2. Dynamics of neuron polarization as defined by Eq. (5.3). **A**, Polarity versus time. The solid line is a linear fit to the data. **B**, Polarity versus total neurite length. Total neurite length was binned at intervals of 20 μm. The solid line is a Gaussian fit to the data, with an inflection point at 114 μm suggesting a phase transition between unpolarized and polarized states. The nonlinearity of this data agrees with our previous work on the polarization of neurons with exactly two neurites [38]).

5.2 Neuron isolation and culture

Primary hippocampal neurons from embryonic day 18 (E18) Sprague-Dawley rats were used for both time-lapse studies. All animal work was approved by the MIT Committee for Animal Care (which performs annual reviews of all animal protocols) and the Division of Comparative Medicine, and abides by all institutional, state, and federal guidelines for animal welfare. Timed pregnant female rats were purchased from Charles River Laboratories. To maximize consistency for embryo age, tissue harvesting was always performed at the same time of day. The rats were euthanized by CO_2 asphyxiation, followed by cervical dislocation. Male and female E18 rat embryos were removed from the mother, and decapitated with sharp scissors. After decapitation, the embryo's skull was cut and removed prior to scooping out the brain and placing in ice-cold HEPES buffered HBSS in a petri dish surrounded by ice. Meninges were removed prior to dissecting the hippocampi.

Hippocampi were dissected and placed in ice-cold Hank's balanced salt solution (HBSS) buffered with 10 mM HEPES at pH 7.3. The hippocampi were then transferred to and mixed in a solution containing 20 units of papain per ml of HBSS, 1 mM $CaCl_2$, and 1 mM L-cysteine, and then incubated at 37°C for 30 min. The cells were subsequently washed in Neurobasal-B27 (Invitrogen) containing 2 mM glutamine and 100 units/ml penicillin/streptomycin.

For both time-lapse imaging and immunocytochemical studies, prior to cell plating, several wells of two 96-well plates (Matrical) were incubated in 10 µg/ml poly-D-lysine (PDL) in phosphate buffered saline (PBS) overnight at room temperature, rinsed with PBS, incubated in 10 µg/ml laminin in PBS for 1 h at 37°C, and finally rinsed again with PBS. Neurons were plated at a density of ~1000 cells/well in 300 µl of the Neurobasal-B27 media. For the western blot, 1 million cells were plated onto 40 mm glass-bottom dishes that were similarly coated with PDL and laminin.

5.3 Synchronous symmetry breaking in neurons with different neurite counts

We experimentally determined whether neurons with different neurite counts polarize synchronously or asynchronously. We measured neurite lengths and polarities in two hundred E18 rat

hippocampal neurons, cultured on glass coated with poly-D-lysine and laminin, at nine different time points over the course of two days. The specific numbers of neurons at each time point are shown in Table 5.1.

Table 5.1. Summary of the number of neurons and neurite data at each time point.

Time point (h)	Number of neurons	Neurite count (mean ± SE)	Polarity (mean ± SE)
7.5	70	2.94 ± 0.13	0.217 ± 0.017
12	94	3.29 ± 0.13	0.233 ± 0.015
17	111	4.11 ± 0.16	0.237 ± 0.013
22.5	113	4.68 ± 0.19	0.248 ± 0.013
28.5	122	4.96 ± 0.21	0.253 ± 0.014
34.5	125	5.40 ± 0.22	0.289 ± 0.015
40.5	125	5.82 ± 0.24	0.295 ± 0.015
46.5	124	6.10 ± 0.25	0.313 ± 0.014
52.5	122	6.60 ± 0.25	0.341 ± 0.015

For this time-lapse experiment, the locations of the 200 randomly chosen neurons were recorded after 3.5 h *in vitro* so that subsequent images of these neurons could be taken quickly. A total of nine time points of images were taken, at 7.5, 12, 17, 22.5, 28.5, 34.5, 40.5, 46.5, and 52.5 h after the neurons were plated. Neurons were discounted if no neurites grew out over the two days, or if there was significant crossing of the neurites. Time points from specific neurons were also discounted if their neurites came into contact with the neurites of neighboring neurons such that the neurites of two neurons could not be distinguished. Of the 200 neurons, 132 were used for further analysis.

Images were captured using a 20× objective (NA 0.75, Nikon) and a CoolSnap HQ2 CCD camera via bright field microscopy. To enhance the contrast of the neurons' boundaries, an adaptive histogram equalization algorithm (*adapthisteq* in MATLAB v7.12) was applied to the images. Skeletons of the neurites were identified manually. When a neurite branched, the longest branch was used in measuring the neurite's length.

At each time point, with the lone exception at 40.5 h after plating, we observed no statistical difference among the polarities of neurons with different neurite counts ($p > 0.05$ by ANOVA, Figure 5.3). We also found that there was no significant correlation ($p > 0.01$ by Pearson correlation) between neurite count and polarity at any of the time points, including the 40.5 h time point. Our results show that neurons with different neurite counts polarize synchronously.

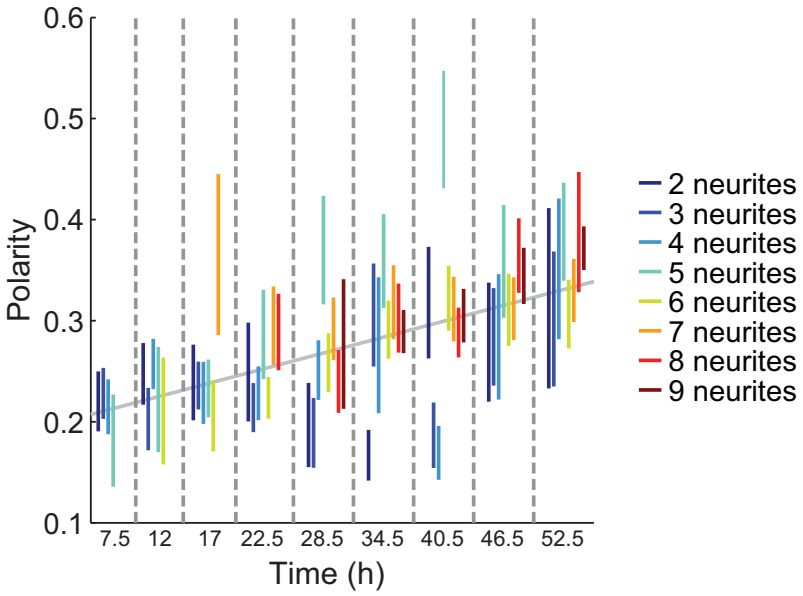

Figure 5.3. Experimentally measured symmetry breaking in neurons with different neurite counts. Each color series indicates a different neurite count, as listed in the legend. Dashed lines separate data from different discrete time points. Bars indicate mean plus/minus SE, and neurite counts at each time point are only shown if at least 3 neurons had that neurite count at that time point. The solid trend line is the same as in Figure 5.2A.

We also verified that the neurons were functionally polarizing over this time scale by performing an immunocytochemical stain for axonal and dendritic markers (Figure 5.4). For this stain, A batch of E18 hippocampal neurons was cultured for 40 h and stained for tau1 and MAP2, which are axonal and dendritic markers, respectively [66]. For tau1, the primary antibody used was mouse monoclonal (MAB3420, Millipore). For MAP2, the primary antibody used was rabbit polyclonal (AB5622, Millipore).

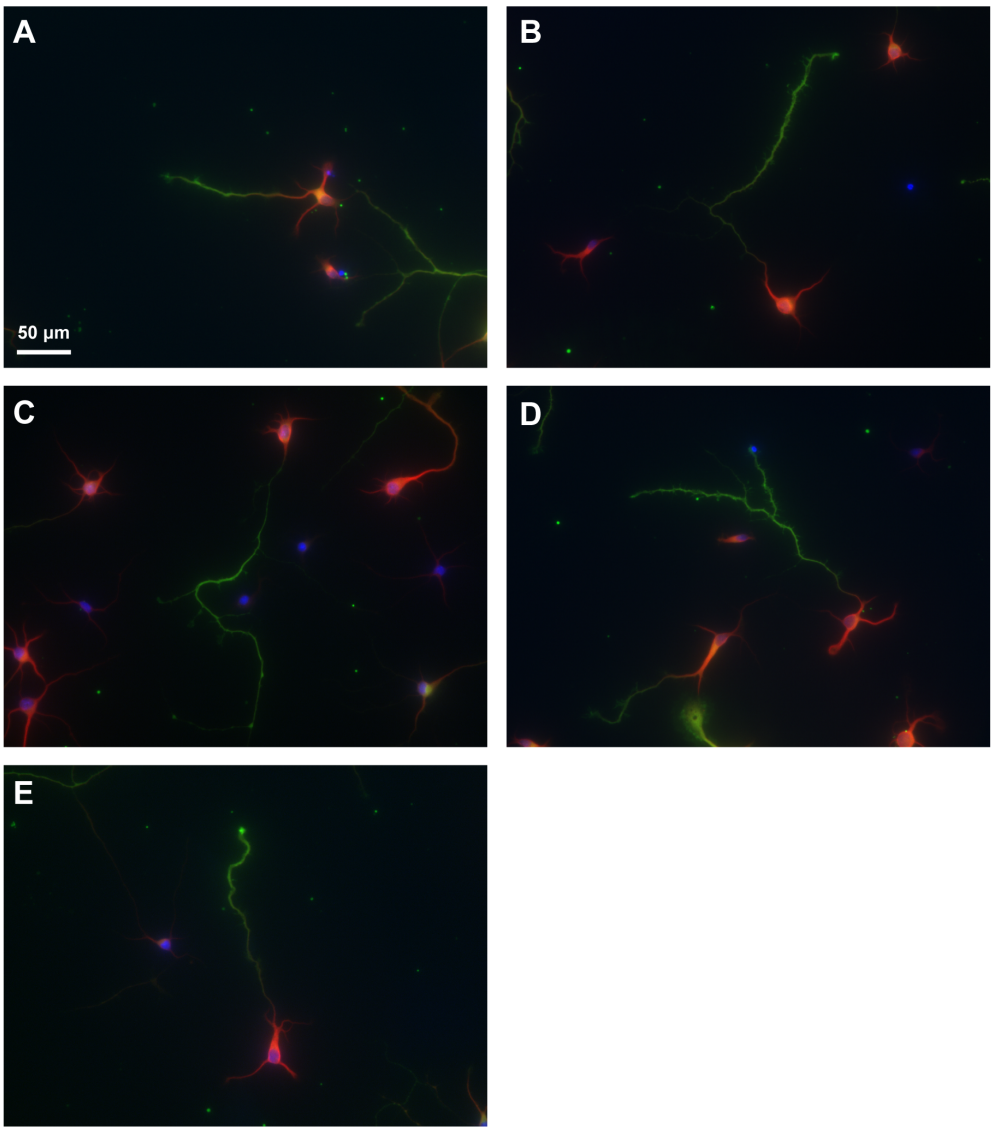

Figure 5.4. Fluorescent immunocytochemical stain for axonal markers 40 h after plating. **A**-**E** are five different micrographs of representative neurons. Tau1, an axonal marker, is shown in green, while MAP2, a dendritic marker, is shown in red. Nuclei were stained with DAPI, and are shown in blue.

5.4 Predicted symmetry breaking rates

In this section, we discuss the predictions made by the Samuels, Fivaz, and Toriyama models regarding how neurite count affects the rate of symmetry breaking in developing neurons. None of the authors of the original models performed such an analysis in their original works. The authors also assumed that neurite count was static during polarization, which we experimentally found to not be true, and which we address in Section 5.2.4.

While the Samuels and Toriyama models predict that neurons with many neurites polarize up to several days slower than neurons with only 2 or 3 neurites, the Fivaz model disagrees, predicting instead that neurons polarize at the same rate, regardless of their neurite counts.

For each of the models we analyzed, 5% Gaussian noise was added to all initial conditions, consistent with the initialization routine of Fivaz et al. [9]. Due to inherent differences among the models, the simulation duration for each model was set so that the polarity of neurons with exactly two neurites achieved a polarity as defined by Eq. (5.3) of approximately 0.5. This ensured that, independent of other model parameters that may affect polarization rate, we could compare the effect of neurite count on polarization dynamics. The Samuels model was run for at least 50 h, the Fivaz model for at least 6 h, and the Toriyama model for at least 130 h. For all three models, neurites were initialized to a length of 5 μm, prior to adding Gaussian noise. For each model, 20 simulations were run for each neurite count. The Samuels model was solved using the MATLAB ordinary differential equation solver *ode45*, while the more complex Fivaz and Toriyama models were solved using the MATLAB function *ode15s*.

5.4.1 Samuels model

The Samuels model consists of three dimensionless parameters, which were set to values that induce axon specification ($\chi_1 = 5$, $\chi_2 = 100$, and $\chi_3 = 5$), as well as a characteristic length (set to 50 μm) and time, which was set to either 20 h (when neurite count was fixed) or 80 h (when neurite count was dynamic) so that the different simulations broke symmetry on similar timescales and could be compared [8]. Prior to Gaussian noise, the initial non-dimensionalized concentration of the rate-limiting protein in the cell body was set to $1/\chi_2$, its steady-state value, and the concentration in each neurite tip was set to 1 divided by the neurite count (the steady-state value when there is no axon specification).

In the Samuels model, as neurite count increases, symmetry breaking slows dramatically (Figure 5.5). Within the first 50 h of the simulation, only neurons with 2 or 3 neurites have begun to polarize (Figure 5.5A). Indeed, neurons with any neurite count do polarize after a sufficient time (Figure 5.5B), but this time grows dramatically with neurite count (Figure 5.5C). For example, a neuron with 9 neurites will polarize approximately 17 times slower than a neuron with only 2 neurites.

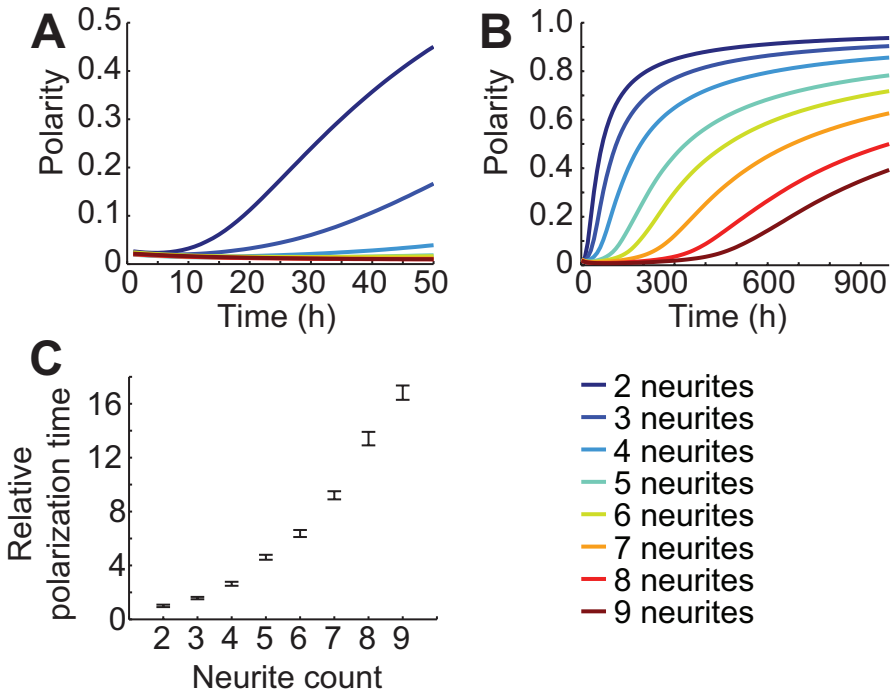

Figure 5.5. Neuronal symmetry breaking as a function of neurite count in the Samuels model. **A**, Polarity versus time for different neurite counts over the first 50 hours of simulation. **B**, Polarity versus time for different neurite counts over the first 900 hours of simulation. **C**, Relative time at which neurons with different neurite counts achieved a polarity of 0.3 Error bars show mean ± SE.

5.4.2 Fivaz model

For the Fivaz model, all parameters were set equal to those originally proposed . Fivaz and coworkers initialized the quantity of HRas and phosphatidylinositol-3,4,5-triphosphate in each neurite to a constant value, resulting in an approximately linear relationship between neurite count and HRas quantity [9].

In direct contrast to the Samuels model, in the Fivaz model the rate of symmetry breaking does not depend on neurite count (Figure 5.6). Neurons with neurite counts ranging from 2 to 9 all appear to break symmetry at the same rate ($p > 0.05$).

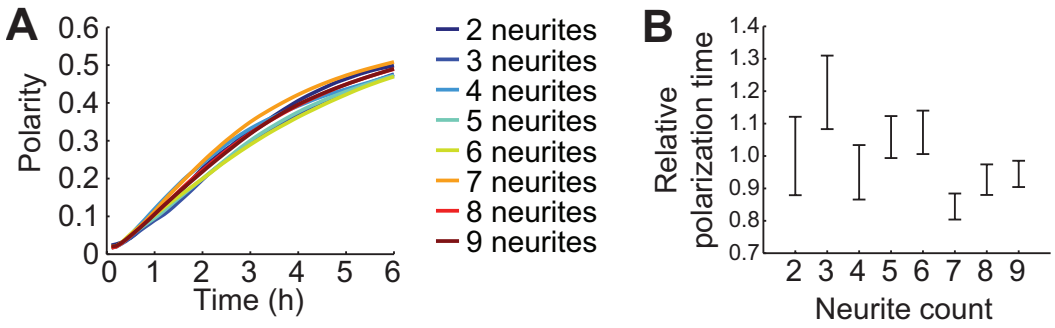

Figure 5.6. Neuronal symmetry breaking as a function of neurite count in the Fivaz model. **A**, Polarity versus time for different neurite counts over the first 50 hours of simulation. **B**, Relative time at which neurons with different neurite counts achieved a polarity of 0.3 Error bars show mean ± SE.

One of the key differences among the Samuels, Fivaz, and Toriyama models is that the Fivaz model predicts that HRas (the key protein involved in symmetry breaking in the Fivaz model) expression increases with neurite count, while the Samuels and Toriyama models have no such stipulation for any of their proteins of interest.

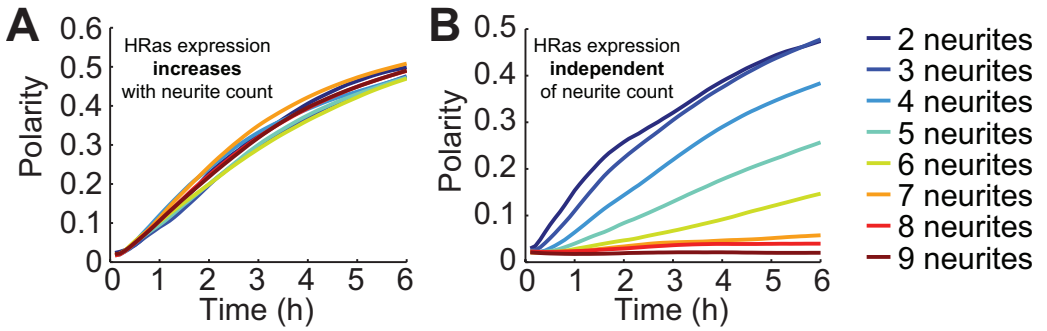

Figure 5.7. Neuronal symmetry breaking as a function of neurite count in the Fivaz model in which HRas expression either increases or remains constant as neurite count increases. **A**, As in Figure 5.6, symmetry breaking occurs synchronously for different neurite counts in the original Fivaz model. **B**, When the Fivaz model is modified so that HRas expression is normalized to the same value for different neurite counts, the Fivaz model then behaves similarly to the Samuels model, in that neurons with more neurites polarize much more slowly that neurons with fewer neurites.

To determine whether this increase in HRas expression was necessary for synchronous symmetry breaking in the Fivaz model, we normalized HRas expression so that it was equal for neurons with different neurite counts (Figure 5.7). To do this, we normalized the initial HRas and phosphatidylinositol-3,4,5-triphosphate (PIP_3) concentrations so that the total quantity of HRas in the cell body and neurite tips

remained constant for each neurite count. For these neurons, those with fewer neurites polarized much faster than neurons with many neurites, as in the Samuels model.

5.4.3 Toriyama model

The Toriyama model was simulated largely as originally described [11] (see Section 1.3.4 for the full list of equations in the Toriyama model). However, we were unable to produce axons using the original equation for the somatic shootin1 concentration, S:

$$S = V_0 \left[\frac{1.9}{1 + \left(t / 3199 \right)^{-2.5}} + 0.07 \right]. \tag{5.5}$$

In this equation, V_0 is the volume of the cell body, and t is time in minutes. To produce neurons with single axons, we increased the numerator in the time-dependent term by approximately 50 percent, to 2.7:

$$S = V_0 \left[\frac{2.7}{1 + \left(t / 3199 \right)^{-2.5}} + 0.07 \right]. \tag{5.6}$$

This model is unique among the three in that symmetry breaking is not evident until dozens of hours into the simulation, even when only two neurites are present.

In their original paper, Toriyama et al. show that neurons with different neurite counts are all able to break symmetry [11]. However, they do not call attention to their findings that neurons with more neurites break symmetry hours or even days slower, although this finding is immediately transparent in their supplementary results (Figure 5.8).

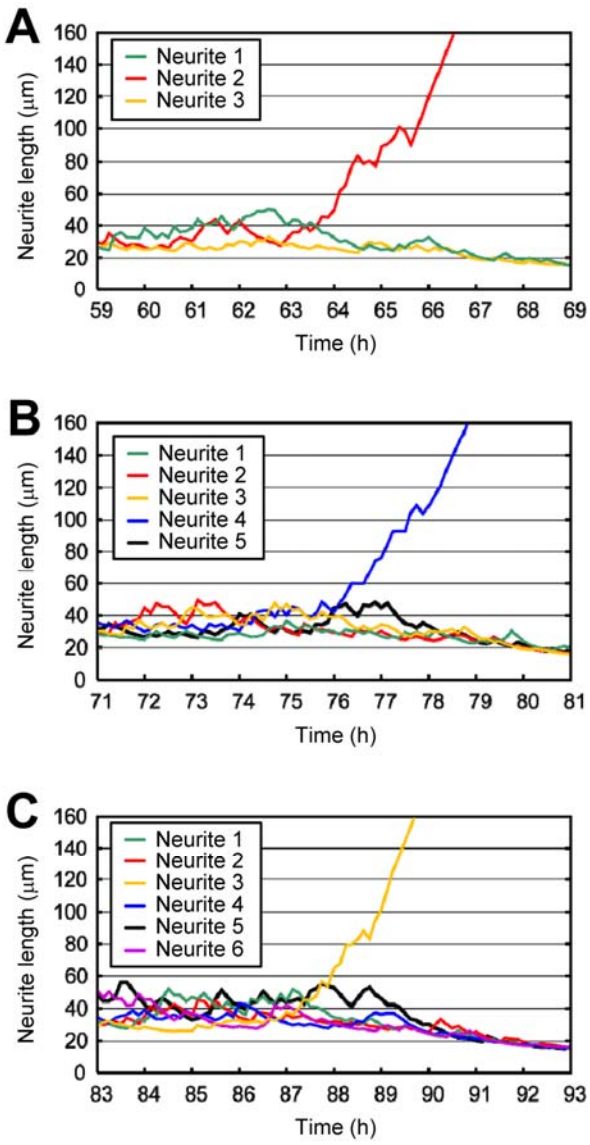

Figure 5.8. Symmetry breaking among neurons with different neurite counts, using the Toriyama model, as reported by Toriyama et al. Note that the neuron with 3 neurites breaks symmetry around 65 hours, the neuron with 4 neurites breaks symmetry around 77 hours, and the neuron with 6 neurites breaks symmetry around 89 hours. Taken from [11].

Our own simulations of the Toriyama model agree with Figure 5.8. We found that the Toriyama model behaved similarly to the Samuels model, and to the Fivaz model when HRas expression was independent of neurite count. In the Toriyama model, neurons with more neurites broke symmetry significantly slower than neurons with fewer neurites (Figure 5.9).

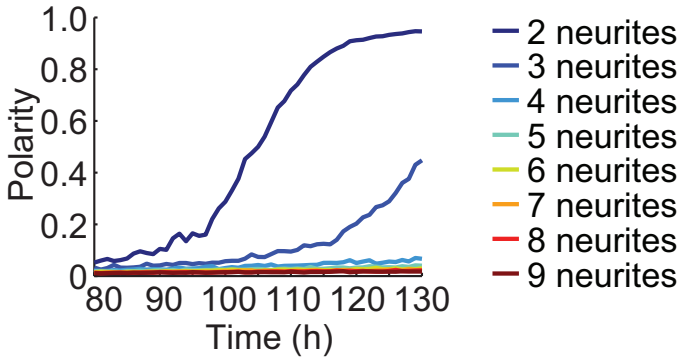

Figure 5.9. Neuronal symmetry breaking as a function of neurite count in the Toriyama model. Symmetry breaking occurs much more quickly in neurons with fewer neurites.

5.4.4 Dynamic neurite count

The Samuels, Fivaz, and Toriyama models all assume that neurite count does not change throughout the polarization process. However, we observed that new neurites sprouted while the neurons were breaking symmetry. We specifically found that neurite sprouting occurred in an exponentially decaying fashion, with a characteristic time of approximately 32 h and an asymptotic mean neurite count of approximately 8.0 (Figure 5.10A).

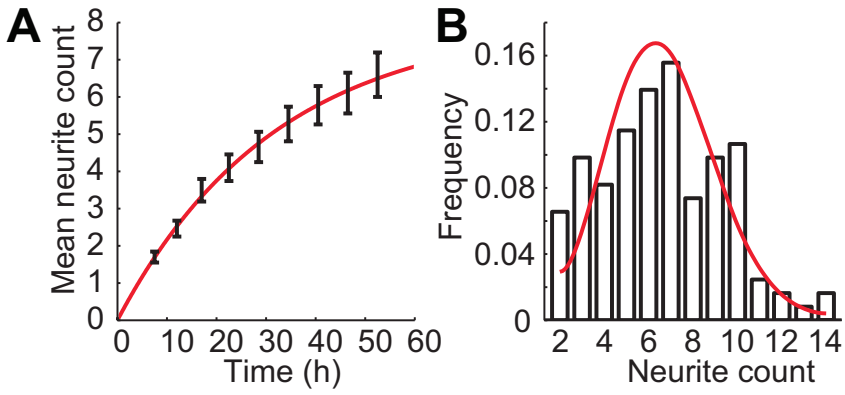

Figure 5.10. Neurite sprouting in developing neurons. **A**, Mean neurite count as a function of time. The solid line is an exponential fit to the data, constrained to include the origin. Error bars are SE. **B**, Neurite count distribution at the final measured time point 52.5 h after plating. The solid line is a simulated distribution using the exponential fit from **A** to calculate the sprouting rate and assumes random neurite sprouting independent of neurite count.

We further analyzed the neurite count distribution at the final measured time point 52.5 h after plating (Figure 5.10B). This distribution was consistent with random neurite sprouting that occurred at a

rate obtained from the exponential fit in Figure 5.10A and that was independent of neurite count (reduced $\chi^2 = 1.5$).

We then modified the Samuels, Fivaz, and Toriyama models by including this "dynamic neurite count," thereby making them more biophysically accurate. We hypothesized that adding a dynamic neurite count to the models could make their symmetry breaking more synchronous, if the models' neurons polarized when they had relatively fewer neurites, and then grew more neurites at a later time.

When neurite count was dynamic, the neurite count was initialized to one, and new neurites were added to the model with an exponentially decaying probability, an initial length of 5 μm, and zero concentration of all model-specific molecules (e.g., HRas, shootin1, etc.), except where otherwise noted. In the Fivaz model with HRas expression independent of neurite count, sprouting neurites were initialized with zero HRas and PIP_3 so that expression levels remained fixed when neurite count was dynamic. The amplitude for the exponential sprouting rate was based on the fit to our experimental data, which had an asymptotic neurite count of 8 (see Figure 5.10A). The time constant was similarly based on our experimental measurement of approximately 32 h, but was scaled to the model-specific simulation durations.

The addition of a dynamic neurite count increased the polarity of neurons with many neurites in the Samuels model (Figure 5.11A) and in the version of the Fivaz model in which HRas expression was independent of neurite count (Figure 5.11B) (compare with Figures 5.3A and 5.5B). However, polarization among neurons with different neurite counts remained asynchronous, as the separation between the polarity vs. time curves remained significant. Adding a dynamic neurite count also had little effect on the Toriyama model (Figure 5.11C), except to universally delay symmetry breaking in all neurons (compare with Figure 5.9). In summary, we found that a dynamic neurite count was insufficient for explaining the synchronous polarization behavior we observed in neurons with different neurite counts.

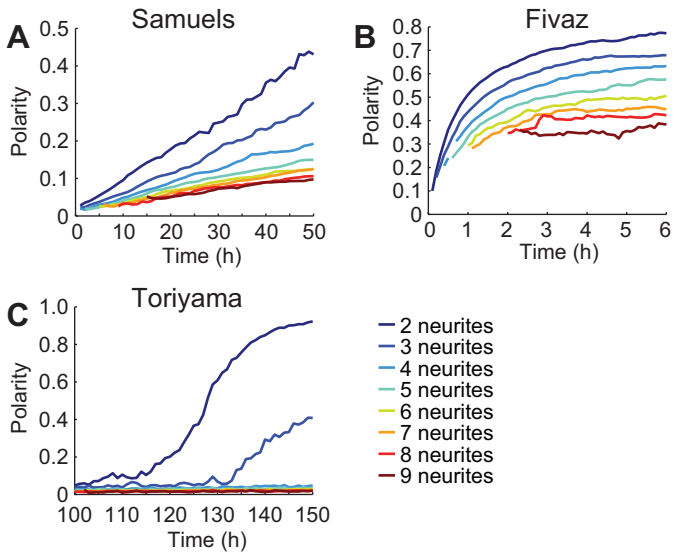

Figure 5.11. Polarity as a function of time and neurite count as predicted by the Samuels (**A**), Fivaz (**B**), and Toriyama (**C**) models of neuronal symmetry breaking modified with dynamic neurite counts. Continuous curves were generated by connecting data for each neurite count at different time points. In the Fivaz model, HRas expression was normalized so that it was independent of neurite count, as in Figure 5.7B.

5.5 Increasing HRas/shootin1 expression explains synchronous symmetry breaking in the models

Among the Samuels, Fivaz, and Toriyama models with both fixed and dynamic neurite counts, the only potential explanation we found for the synchronous polarization of neurons with different neurite counts was that the HRas expression levels in the Fivaz model increased with neurite count. In the Fivaz model, neurite growth and polarization rates ultimately depend on HRas concentration by way of several positive feedback loops, and so an increase in HRas levels can accelerate neurite growth and axon specification.

The Toriyama model consists of similar feedback loops that depend on shootin1 expression levels. We hypothesized that if shootin1 expression levels increased with neurite count in the Toriyama model, and similarly that if expression levels for the rate-limiting chemical for neurite growth increased with neurite count in the Samuels model, then both of these models might also agree with our experimental observations in Figure 5.3.

In the Samuels model, the original equation for the expression of the rate-limiting chemical for neurite growth in the cell body is:

$$\frac{dC_0}{dt} = 1 - \sum_{i=1}^{N} T_i \,, \tag{5.7}$$

where C_0 is the normalized concentration in the cell body, N is the neurite count, and T_i is the active transport rate to neurite i. We modified this equation so that protein expression was modeled as being proportional to neurite count:

$$\frac{dC_0}{dt} = N - \sum_{i=1}^{N} T_i \,. \tag{5.8}$$

In the Fivaz model with a dynamic neurite count and in which HRas expression increased with neurite count, sprouting neurites were initialized with the same HRas and PIP_3 expression levels that were used when initializing the model.

To model protein expression that increases with neurite count in the Toriyama model, we modified Eq. (5.6) similarly to how we modified the Samuels model, i.e., by adding a linear term that depends on neurite count N:

$$S = V_0 \left[\frac{1.9\left(1 + N/5\right)}{1 + \left(t/3199\right)^{-2.5}} + 0.07 \right]. \tag{5.9}$$

The results of these modifications are shown in Figure 5.12. The polarity vs. neurite count curves are all closely bundled together for each model. Thus, by coupling the expression level of the protein underlying symmetry breaking (e.g., HRas, shootin1, or some other rate-limiting protein for neurite growth) to neurite count, all three models become consistent with our experimental finding that neurons with different neurite counts polarize synchronously.

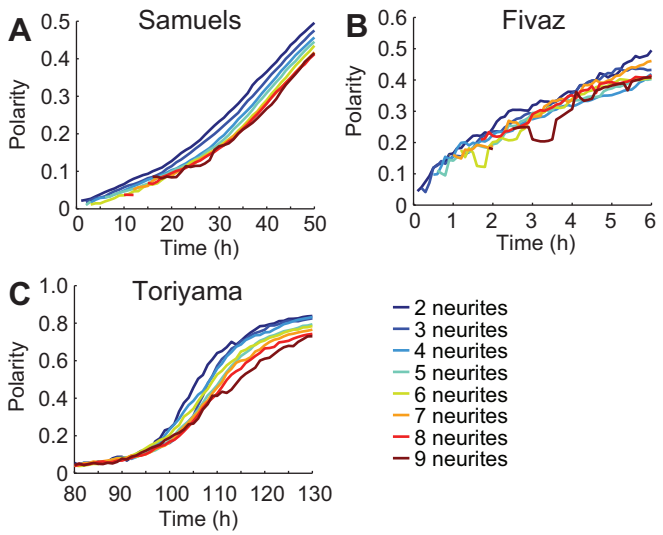

Figure 5.12. Polarity as a function of time and neurite count as predicted by the Samuels (**A**), Fivaz (**B**), and Toriyama (**C**) models of neuronal symmetry breaking modified with dynamic neurite counts. Continuous curves were generated by connecting data for each neurite count at different time points. In the Fivaz model, HRas expression was normalized so that it was independent of neurite count, as in Figure 5.7B.

5.6 Measurement of increasing HRas/shootin1 expression in developing neurons

We then tested our hypothesis that HRas and/or shootin1 expression levels increase with neurite count. To do this, we performed immunocytochemical stains of hippocampal neurons, and then calculated the relative fluorescence of approximately 150 neurons for each stain, recording neurite count as well.

For these stains, a second batch of E18 hippocampal neurons was cultured for 40 h and then fixed for 30 min in 4% paraformaldehyde, and then washed twice using PBS with 0.05% Tween-20 (PBST). The neurons were then permeabilized for 10 min with 0.1% Triton X-100, and washed again in PBST. The surface was next blocked for 30 min with 3% bovine serum albumin (BSA) in PBS. The cells were then incubated for 60 min in primary antibodies: either mouse anti-HRas monoclonal antibody (Millipore) or rabbit anti-shootin1 polyclonal antibody (Pierce). After another wash in PBST, the secondary antibodies were applied for another 60 min: cells stained with mouse anti-HRas were then stained with goat anti-mouse AlexaFluor 555 (Invitrogen) and goat anti-rabbit AlexaFluor 546 (Invitrogen). Both secondary antibodies were visible using a Cy3 filter cube. The cells were again washed in PBST a final time.

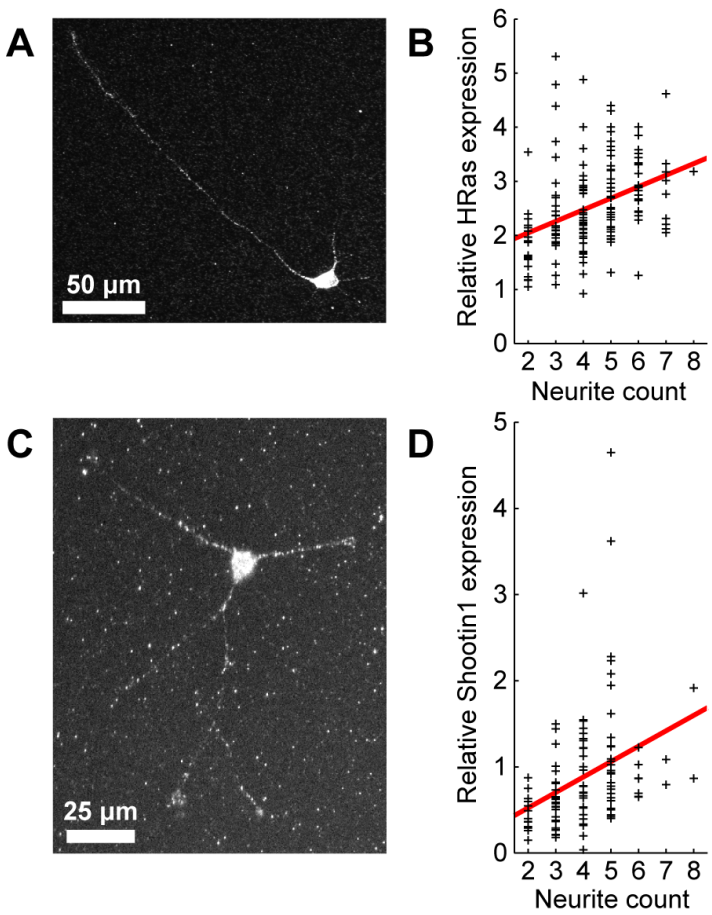

Figure 5.13. HRas and shootin1 expression in developing hippocampal neurons as a function of neurite count after 40 h in culture. **A**, Typical immunocytochemical stain for HRas in a neuron with 5 neurites. **B**, Typical immunocytochemical stain for shootin1 in a neuron with 3 neurites. The brightness in both **A** and **B** were saturated to make the neurites more visible. When these images were analyzed to determine relative HRas and shootin1 expression levels, image brightness was kept unsaturated. **C**-**D**, HRas/shootin1 expression as a function of neurite count. Individual neurons are indicated by plus signs, while the solid line indicates linear fits to the data. In both trend lines, the slope was significantly positive ($p < 10^{-4}$ for both fits).

Immunostained neurons were imaged using the same objective and camera used for the time-lapse recordings. A total of 156 HRas-stained neurons and another 111 shootin1-stained neurons were selected for analysis. Neurons were chosen if they had a clearly defined neurite count and they were not in contact with other neurons. The relative amount of HRas or shootin1 in each of the neurons was determined by cropping a polygon surrounding the entire neuron, including its cell body and all of its neurites, from the fluorescence image. The background pixel brightness was multiplied by the area of the cropped image, and this quantity was then subtracted from the total integrated brightness of the cropped image. Fluorescence

images were also appropriately normalized to account for any inhomogeneities in the periphery of the focal plane due to optical aberrations.

Figures 5.13A-B show sample stains for HRas and shootin1, respectively. We found that both HRas and shootin1 expression significantly increased with neurite count (Figures 5.13C-D, see Table 5.2 for more information).

Table 5.2. Summary of the number of neurons and relative expression levels of HRas and shootin1 as determined by immunocytochemistry.

	HRas		Shootin1	
Neurite count	**Number of neurons**	**Relative expression (mean ± SE)**	**Number of neurons**	**Relative expression (mean ± SE)**
2	19	1.81 ± 0.13	14	0.46 ± 0.05
3	26	2.52 ± 0.20	30	0.63 ± 0.06
4	42	2.37 ± 0.11	26	0.93 ± 0.12
5	35	2.79 ± 0.13	28	1.28 ± 0.19
6	24	2.94 ± 0.12	7	0.85 ± 0.08
7	9	2.84 ± 0.27	2	0.94 ± 0.15
8	1	3.18	2	1.4 ± 0.5

Since we found that both HRas and shootin1 expression levels increase with neurite count, and that neurite count increases with time, we finally determined whether expression of these proteins also increases with time. Toriyama et al. previously showed via a western blot that shootin1 levels increase by approximately an order of magnitude over the first 24 hours after plating [11] (reproduced in Figure 5.14).

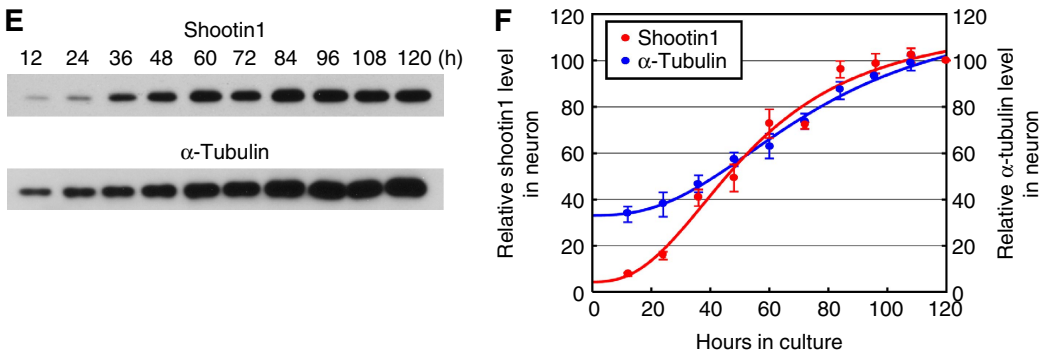

Figure 5.14. Shootin1 (and α-tubulin) expression in developing neurons during the first 120 h after plating, as reported by Toriyama et al. For shootin1, 1.7×10^5 neurons were used, and band densities were measured using a densitometer. Taken from [11].

We performed a similar blot, instead labeling HRas. Our western blot for HRas expression was performed using 5×10^5 neurons at three time points: 3, 7, and 12 h after plating. Cells were lysed using a mammalian cell lysis kit (Sigma), and the lysate was then concentrated using centrifugal filter units (Amicon Ultra 10K, Millipore). The concentrated lysate was run on an SDS-PAGE (4-12% Bis-Tris, Invitrogen), and then transferred to a nitrocellulose membrane under semi-dry conditions. The primary antibody used for the western blot was Rabbit polyclonal to HRas (ab97488, AbCam), and the secondary antibody was tagged with a quantum dot (WesternDot 625 goat-anti-rabbit, Invitrogen) that fluoresced under UV. A UVP GelDoc-It Imager was used for imaging the blot.

The original blot was quite noisy, with significant smearing and optical halo effects from the brightest ladder bands (Figure 5.15A). By subtracting background vertically, we were able to highlight the bands and get a cleaner image (Figure 5.15B). We found that HRas expression significantly increases over the first 12 hours after plating the neurons (Figure 5.16).

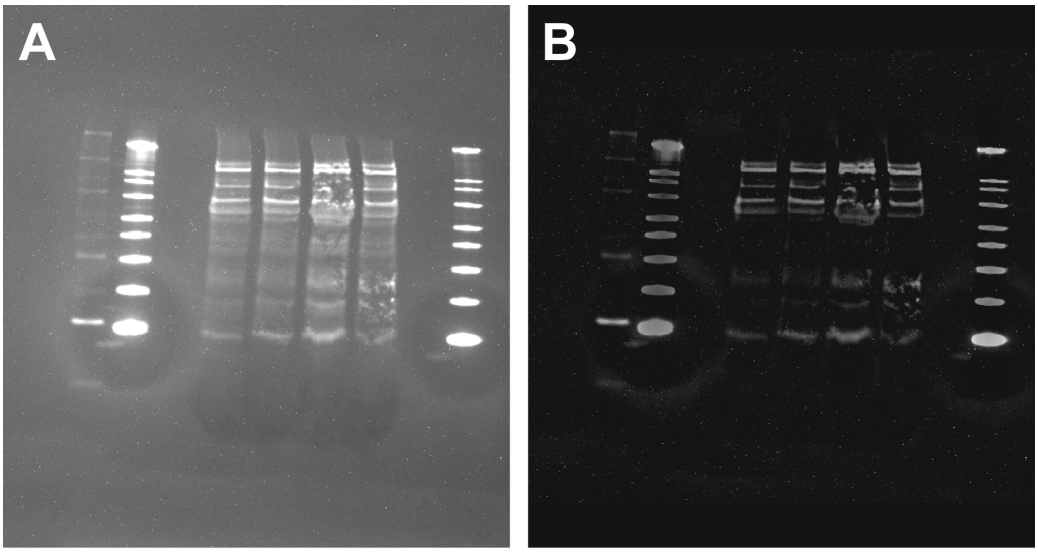

Figure 5.15. Raw (**A**) and enhanced (**B**) western blot iamges for HRas in developing neurons. The enhancement was performed by subtracting background vertically. The bottom-most row of the ladder (right-most lane in **A** and **B**) is 20 kDa.

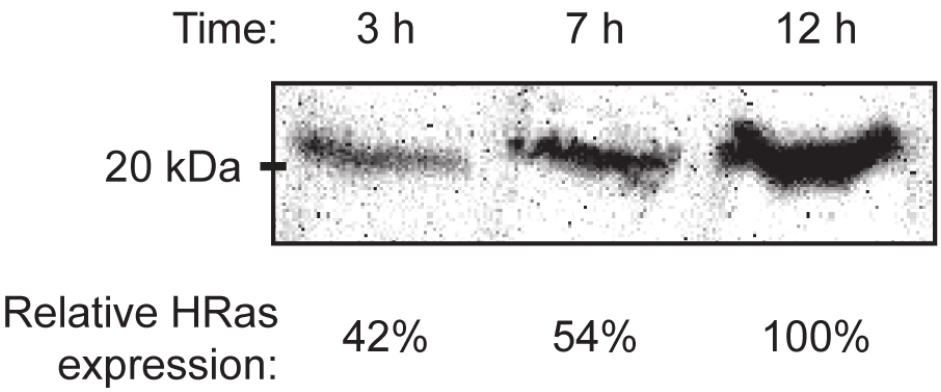

Figure 5.16. Western blot for HRas in developing neurons during the first 12 h after plating. HRas was immunoblotted using a polyclonal antibody 3, 7, and 12 h after plating. Relative HRas expression was quantified by integrating the 20 kDa bands in each lane.

5.7 A mechanism for synchronous symmetry breaking in neurons with different neurite counts

Here we have examined, for the first time, the role of neurite count in neuronal symmetry breaking. To do this, we introduced in Section 5.1 a new metric for neuronal polarity that uses information about all of a neuron's neurites, unlike previously used metrics [9,63–65]. Other metrics that meet this criterion are possible – for example, one might construct a polarity metric based on Shannon entropy, treating the relative lengths of the neurites as discrete probabilities. However, the metric we used is relatively simple, captures the phase transition as neurons polarize (Figure 5.2B), and successfully illustrates the similarities and differences among various models and our experimental data. Furthermore, because of our metric's linear nature, it is better suited for measuring small changes in polarity when neurons are highly unpolarized.

We then applied this metric to study whether neurons with different neurite counts polarize at different times. Since a neuron with more neurites should have more degrees of freedom during symmetry breaking, one may expect to find that neurons with more neurites take longer to polarize. For instance, Toriyama et al. reported that their model predicts a 24 h temporal lag between symmetry breaking in neurons with 3 and 6 neurites [11]. In both the Samuels and Toriyama models, we found that neurons with 10 neurites polarize several days later than neurons with 2 or 3 neurites do. However, previous experiments had shown that neurons with various final neurite counts all polarize within the same 48 h time

window [3,61,62]. Our experiments, which were conducted with a temporal resolution of 6 h, were consistent with these findings: we found no time delay in the polarization of neurons with different neurite counts, and that they instead broke symmetry synchronously. Such coordinated symmetry breaking may be important for synchronized polarization of neurons in developing organisms.

The leading models agree that additional neurites should slow symmetry breaking when the expression of specific proteins (e.g., HRas in the Fivaz model, shootin1 in the Toriyama model, or some other rate-limiting protein for neurite growth in the Samuels model) is independent of neurite count. We have shown that increased expression of these proteins accelerates symmetry breaking in each model. This acceleration occurs because the models include these proteins in positive feedback loops, so that increases in their intracellular concentration speed up the entire polarization process. When the expression of these proteins increases with neurite count, the two effects (increased protein expression accelerating symmetry breaking, a higher neurite count decelerating symmetry breaking) can cancel out, so that neurons with different neurite counts break symmetry at the same rate, consistent with our experimental results.

Such a relation between protein expression and neurite count does not inherently contradict the previous experimental findings of Fivaz et al. or Toriyama et al. The Fivaz model already assumed that HRas expression and neurite count were proportional, while Toriyama et al. measured overall shootin1 expression in bulk, using 1.7×10^5 neurons, and so it is possible that shootin1 expression varied with neurite count in their experiments. Here, we measured HRas and shootin1 expression levels in several hundred developing neurons individually, and found that the expression of both proteins increased significantly with neurite count.

Future work might examine the causality of these trends, i.e., whether a higher neurite count increases HRas/shootin1 expression, or whether higher HRas/shootin1 expression levels increase neurite count. HRas and shootin1 overexpression have been previously shown to induce the formation of supernumerary axons [9–11], but have not yet been shown to affect neurite count. Alternatively, both HRas/shootin1 expression levels and neurite sprouting could be controlled by some other agent, such as intrinsic proteins (e.g., Rho GTPases, the PAR complex, etc.) or extrinsic signaling [4].

Thus, while additional neurites could hinder symmetry breaking, our results suggest that neurons can and do overcome this via increased expression levels of their symmetry-breaking machinery (e.g.,

HRas, shootin1, etc.), allowing neuronal symmetry breaking among neurons with different neurite counts to occur synchronously.

Chapter 6

Conclusions

This thesis represents the first experimental comparison of biophysical models developed to explain symmetry breaking in neuronal development (in Chapters 4 and 5), as well as the first examination of the role of neurite count in neuronal polarization (in Chapter 5).

Future studies of neuronal symmetry breaking could explore a number of additional specific questions. For example, how is the correlation we observe between neurite count and shootin1/HRas expression maintained? Four possible explanations are that (1) increased expression of these key proteins also induce neurite sprouting, (2) neurite sprouting somehow induces increased expression of these proteins, (3) some external mechanism regulates both the protein expression and neurite count so that they increase together, or (4) neurite count and the expression of key proteins happen to increase over the same time scale in such a way that symmetry breaking is synchronous for different neurite counts. Of these, we suspect that (2) is unlikely, since it would require the cell body to maintain a "sense" of neurite count. The remaining possibilities, (1), (3), and (4), all seem plausible, and so further experimentation would be necessary to explain our observed correlation.

One such experiment could involve overexpression or underexpression of the key proteins. For example, if RNA interference against HRas results in neurons with fewer neurites, then that would be evidence for explanation (1). Alternatively, one could attempt to control neurite count in stage II and stage III neurons and measure any changes in shootin1/HRas expression. In such experiments, neurite count could be controlled either through axotomy or by growing neurons on a patterned substrate, such as a hexagonal array of polylysine printed onto a poly(ethylene glycol) surface (as described in Chapter 2).

Another study could examine the signaling mechanism between shootin1 and HRas. Toriyama et al. acknowledge the work of Fivaz et al. when describing their own model [9,11], although the two models have not yet been unified. Does shootin1 aggregation in a neurite tip responsible for an increase in HRas transport to that tip? Experiments using molecular biology techniques, such as fluorescent resonance energy transfer (FRET) imaging, might help to resolve this question.

The LAMP system for rapid protein patterning we describe in Chapter 2 can also be applied to a host of studies in cellular biology and neuroscience. If it were possible to precisely position individual neurons onto specific points (e.g., in a rectangular array) on the pattern, then these studies could be more readily performed. The LAMP system is particularly well suited to experiments on axon guidance, i.e., the directed migration of axonal growth cones via chemical and physical gradients and cues. By patterning simultaneous arbitrary gradients of different signaling molecules, such as nerve growth factor, and proteins from the netrin, ephrin, and semaphorin families, the combined effects of these guidance cues could be studied. If properly controlled, the patterning of these cues could be extended to create not just arbitrary protein gradients, but arbitrary neural networks. Patterning proteins that induce synaptogenesis (e.g., neurexin/neuroligin) could control where synapses form between neurons.

To our knowledge, the phenomenon of "synchronous symmetry breaking," in which a system has a built-in mechanism for overcoming additional complexity when breaking symmetry, has never been previously described. Synchronous symmetry breaking may very well occur in other biological systems with coordinated symmetry breaking, especially during development.

Finally, subtle abnormalities in the processes of neuronal development (e.g., symmetry breaking, neurite growth, axon guidance, synaptogenesis, etc.) have been implicated in several neurological disorders such as autism [39,40], schizophrenia [41], and epilepsy [42]. A better understanding of how neuronal development occurs, including the molecular and biophysical mechanisms behind the abovementioned processes, could ultimately lead to better treatments for these disorders.

Appendix A

MATLAB Code for Models of Neuronal Symmetry Breaking

This appendix contains original MATLAB code that we wrote for simulating the Khanin [7], Samuels [8], Fivaz [9], and Toriyama [11] models. The code for the Samuels, Fivaz, and Toriyama models was specifically used for simulating the polarity of neurons with different neurite counts.

As far as we know, this is the first and only collective description of these molecular neuronal symmetry breaking models.

A.1 Khanin model

```
% Basic MATLAB program for simulating the Khanin model.

N = 5;  % neurite count
alpha = 1.2;  % strength of competition in the Khanin model

tmax = 100;    %simulation duration

L = zeros(tmax,N);  % array of neurite lengths
L(1,:) = ones(1,N);

for i = 2:tmax

    x = rand(1);

    % construct array of probabilities that each nuerite will grow in
the
    % next time step
    p_array = L(i-1,:).^alpha/sum(L(i-1,:).^alpha);

    % construct array of cumulative probabilities
    cump_array = zeros(N,1);
    for j = 1:N
        cump_array(j) = sum(p_array(1:j));
    end

    L(i,:) = L(i-1,:);

    k = find(x < cump_array);
    L(i,k(1)) = L(i,k(1))+1;

end

figure(1);
```

```
clf;
plot(L);
```

A.2 Samuels model

```
% Basic MATLAB program for simulating the Samuels model. Determines the
% polarity for different neurite counts after various times.

% Name of the ODE function program
odefun = 'samuels_plain_ODE';

% number of iterations
num_iter = 20;

time_array = 1:1:50;

tmax = max(time_array);

global P;
global NN;

P = zeros(1,5);

% Parameters of the model

% time_scale
P(1) = 20;
% length_scale
P(2) = 50;
% x_1
P(3) = 5;
% x_2
P(4) = 100;
% x_3
P(5) = 5;

% Array of neurite counts to test
NN_array = 2:9;

% Matrix containing the polarities vs. time data for each simulation
polarity_matrix = zeros(length(NN_array),length(time_array),num_iter);

h = waitbar(0);

for nn = 1:length(NN_array)

    % NN = neurite count
    NN = NN_array(nn);

    for z = 1:num_iter

        % Beginning of Samuels model ----------------------
```

```
        C0 = 1/P(4);
        C = ones(NN,1)/NN;
        L = 5*ones(NN,1);

        g = 0.05;
        P_local = [C0;C;L];
        P_local = P_local .* (1+g*randn(size(P_local)));

        y0 = P_local';

        [t y] = ode45(odefun,[0 tmax],y0);

        % End of the Samuels model ----------------------

        % determine relative neurite lengths
        NL = y(:,NN+2:end);
        rel_lengths = NL./repmat(sum(NL,2),1,size(NL,2));

        % find polarity of neuron
        polarity_array = NN/(2*(NN-1))*sum(abs(rel_lengths-1/NN),2);

        % update polarity_matrix with the relevant timepoint
        for i = 1:length(time_array)
            throwaway = find(t<time_array(i));
            polarity_matrix(nn,i,z) = polarity_array(throwaway(end));
        end

        waitbar(z/num_iter,h,[num2str(NN) ' neurites']);

    end

end

close(h);

% find mean polarity data for each time point
M = mean(polarity_matrix,3);

% plotting the data
figure;
clf;
hold on;
jet = colormap('jet');
max_neurites = NN_array(end);
xlabel('Time (h)');
ylabel('Polarity');
for i = 1:size(M,1)
    jet_row = max(round((i-1)/(max_neurites-2)*size(jet,1)),1);
    plot(timey_array,M(i,:),'Color',jet(jet_row,:),'LineWidth',2);
end
set(gca,'YTick',0:0.1:0.5);
axis([0 50 0 0.5]);

% ODE for the Samuels model
```

```
function dy = samuels_ODE(t,y)

global P;
global NN;

dy = zeros(2*NN+1,1);

% Parameters of the model
Ts = P(1);
Ls = P(2);
x1 = P(3);
x2 = P(4);
x3 = P(5);

% Transport equation
T = x1*(y(1)-y(2:NN+1))*Ls./y(NN+2:end) + x2*y(2:NN+1)*y(1);

% Soma concentration
dy(1) = 1/Ts * (1-sum(T));

% Neurite tip concentration equations
dy(2:NN+1) = 1/Ts * x3*(T-y(2:NN+1));

% Neurite length equations
dy(NN+2:end) = Ls/Ts * y(2:NN+1);
```

A.3 Fivaz model

```
% Basic MATLAB program for simulating the Fivaz model. Determines the
% polarity for different neurite counts after various times.

% Name of the ODE function program
odefun = 'fivaz_ODE';

% Global parameters
global NN;
global tmax;
global F;
global v_times_array;

% number of iterations
num_iter = 20;

time_array = 0.1:0.1:6;
tmax = max(time_array);

NN_array = 2:9;
polarity_matrix = zeros(length(NN_array),length(time_array),num_iter);

h = waitbar(0);

for nn = 1:length(NN_array)
```

```
    % NN = neurite count
    NN = NN_array(nn);
    v_times = cell(NN,1);

    for z = 1:num_iter

        % Beginning of Fivaz model ---------------------

        % Constants in the model
        f = 60;
        kP = 114.3 * f;
        Rb = 0.005;
        KM = 5;
        phi = 2 * f;
        kR1 = 2.86 * f;
        rho = 2 * f;
        beta = 0.0571 * f;
        Pb = 0.064;
        kR0 = 8.57 * f;

        % HRas-GTP
        GTP = 0.10*ones(NN,1);
        % PIP3
        P = kP/phi.*(GTP+Rb)./(GTP+Rb+KM);
        % HRas-GDP
        GDP = rho/kR1*GTP./(P+Pb);
        % HRas in the soma
        S = beta/kR0*(sum(GDP)+sum(GTP))/(NN*Pb+sum(P));
        % Neurite lengths
        L = 5*ones(NN,1);

        % adding 5 percent random gaussian noise to the initial
conditions
        g = 0.05;
        P_local = [S;P;GTP;GDP;L];
        P_local = P_local .* (1+g*randn(size(P_local)));

        y0 = P_local';

        F = tmax*60;

        % sampling rate
        freq = 600;

        % determination of random vesicle times

        time_array = rand(freq*tmax,NN);
        times = time_array < 60/freq;

        for i = 1:NN
            v_times{i} = find(times(:,i))/freq;
        end
        num_v = length(v_times{1});
        for j = 2:NN
            if length(v_times{j}) > num_v
                num_v = length(v_times{j});
```

```
            end
        end
        v_times_array = -100*ones(num_v,NN);
        for j = 1:NN
            v_times_array(1:length(v_times{j}),j) = v_times{j};
        end

        [t y] = ode15s(odefun,[0 tmax],y0);

        % End of the Fivaz model -----------------------

        % determine relative neurite lengths
        NL = y(:,NN*3+2:NN*4+1);
        rel_lengths = NL./repmat(sum(NL,2),1,size(NL,2));

        % find polarity of neuron
        polarity_array = NN/(2*(NN-1))*sum(abs(rel_lengths-1/NN),2);

        % update polarity_matrix with the relevant timepoint
        for i = 1:length(time_array)
            throwaway = find(t<time_array(i));
            polarity_matrix(nn,i,z) = polarity_array(throwaway(end));
        end

        waitbar(z/num_iter,h,[num2str(NN) ' neurites']);

    end

end

close(h);

% find mean polarity data for each time point
M = mean(polarity_matrix,3);

% plotting the data
figure;
clf;
hold on;
jet = colormap('jet');
max_neurites = NN_array(end);
xlabel('Time (h)');
ylabel('Polarity');
for i = 1:size(M,1)
    jet_row = max(round((i-1)/(max_neurites-2)*size(jet,1)),1);
    plot(time_array,M(i,:),'Color',jet(jet_row,:),'LineWidth',2);
end
set(gca,'YTick',0:0.1:0.5);
axis([0 6 0 0.5]);

% ODE for the Fivaz model

function dy = fivaz_ODE(t,y)

global NN;
```

```
global v_times_array;

global F;
global tmax;

dy = zeros(NN*4+1,1);

% HRas in the soma
S = y(1);
% PIP3
P = y(NN*0+2:NN*1+1);
% HRas-GTP
GTP = y(NN*1+2:NN*2+1);
% HRas-GDP
GDP = y(NN*2+2:NN*3+1);
% Neurite lengths
L = y(NN*3+2:NN*4+1);

% Constants in the model
f = 60;
kP = 114.3 * f;
Rb = 0.005;
KM = 5;
phi = 2 * f;
kR1 = 2.86 * f;
rho = 2 * f;
beta = 0.0571 * f;
Pb = 0.064;
kR0 = 8.57 * f;
kL = 7;

lamda = 0;
KL = 20;

% temporal vesicle width
sig = tmax/F;

% at each time point, v represents the number of vesicles (spread over
% a Gaussian distribution) arriving in the time step
v = zeros(NN,1);


for j = 1:NN
    for i = 1:size(v_times_array,1)
        v(j) = v(j)+tmax/(F*sig*sqrt(2*pi))*exp(-(t-
v_times_array(i,j)).^2/(2*sig^2));
    end
end

% HRas in the soma
dy(1) = beta*(sum(GTP)+sum(GDP))-kR0*S*sum((Pb+P).*v);
% PIP3
dy(NN*0+2:NN*1+1) = kP*(GTP+Rb)./(KM+GTP+Rb)-phi*P;
% HRas-GTP
dy(NN*1+2:NN*2+1) = kR1*(P+Pb).*GDP-(rho+beta)*GTP;
% HRas-GDP
```

```
dy(NN*2+2:NN*3+1) = kR0*(P+Pb)*S.*v-beta*GDP+rho*GTP-kR1*GDP.*(P+Pb);
% Neurite lengths
dy(NN*3+2:NN*4+1) = kL*(P+Pb) - lamda*L./(KL+L);
```

A.4 Toriyama model

```
% Basic MATLAB program for simulating the Toriyama model. Determines
the
% polarity for different neurite counts after various times.

% Name of the ODE function program
odefun = 'toriyama_ODE';

% Global parameters
global NN;
global tmax;
global v_time;
global aj;
global f;
f = 60;

% number of iterations
num_iter = 20;

time_array = 80:1:130;
tmax = max(time_array);

NN_array = 2:9;
polarity_matrix = zeros(length(NN_array),length(time_array),num_iter);

h = waitbar(0);

for nn = 1:length(NN_array)

    % NN = neurite count
    NN = NN_array(nn);

    for z = 1:num_iter

        % Beginning of Toriyama model ----------------------

        % shootin1 concentration in each neurite
        C = zeros(NN,1);
        % Neurite lengths
        L = 5*ones(NN,1);

        % adding 5 percent random gaussian noise to the initial
conditions
        g = 0.05;
        P_local = [C;L];
        P_local = P_local .* (1+g*randn(size(P_local)));

        y0 = P_local';
```

```
        % Picking the vesicle times from a gamma distribution
        k = 4.8;
        theta = 3.4/f;

        gamma_x = 0:0.001:(k*theta+10*theta*sqrt(k));
        gamma_y = gammainc(gamma_x/theta,k);

        v_time = cell(1,NN);
        for i = 1:NN

            the_time = 0;
            while the_time < tmax
                yrand = rand(1);
                P = find(gamma_y >= yrand);
                next_interval = gamma_x(P(1));
                the_time = the_time+next_interval;
                if the_time < tmax
                    v_time{i} = [v_time{i}; the_time];
                end
            end

        end

        % Picking the shootin1 quantity inside each vesicle using
another
        % gamma distribution

        k = 5.3;
        theta = 0.25;

        gamma_x = 0:0.001:(k*theta+10*theta*sqrt(k));
        gamma_y = gammainc(gamma_x/theta,k);

        aj = cell(size(v_time));
        for i = 1:NN
            for g = 1:size(v_time{i},1)
                yrand = rand(1);
                P = find(gamma_y >= yrand);
                aj{i} = [aj{i}; gamma_x(P(1))];
            end
        end


        [t y] = ode15s(odefun,[0 tmax],y0);

        % End of the Toriyama model ----------------------

        % determine relative neurite lengths
        NL = y(:,NN+1:end);
        rel_lengths = NL./repmat(sum(NL,2),1,size(NL,2));

        % find polarity of neuron
        polarity_array = NN/(2*(NN-1))*sum(abs(rel_lengths-1/NN),2);
```

```
        % update polarity_matrix with the relevant timepoint
        for i = 1:length(time_array)
            throwaway = find(t<time_array(i));
            polarity_matrix(nn,i,z) = polarity_array(throwaway(end));
        end

        waitbar(z/num_iter,h,[num2str(NN) ' neurites']);

    end

end

close(h);

% find mean polarity data for each time point
M = mean(polarity_matrix,3);

% plotting the data
figure;
clf;
hold on;
jet = colormap('jet');
max_neurites = NN_array(end);
xlabel('Time (h)');
ylabel('Polarity');
for i = 1:size(M,1)
    jet_row = max(round((i-1)/(max_neurites-2)*size(jet,1)),1);
    plot(time_array,M(i,:),'Color',jet(jet_row,:),'LineWidth',2);
end
set(gca,'YTick',0:0.2:1.0);
axis([80 130 0 1.0]);

% ODE for the Toriyama model

function dy = toriyama_ODE(t,y)

global NN;
global f;

dy = zeros(2*NN,1);

% shootin1 concentration in each neurite
C = y(1:NN);
% Neurite lengths
L = y(NN+1:end);

% Constants in the model
rho = 1/300;
sigma = 0.4/f;
dkON = 0.059 * f;
dkOFF = 0.678 * f;
as = 5.10;
al = 2.39;
Kl = 38.8;
h = 4.30;
Ks = 2.15;
```

```
L0 = 9.94;
ADV = 8.26*f;

t1 = 3199/f;
t2 = 4480/f;

C0 = (2.7/(1+(t/t1)^-2.5) + 0.07) - 0.1*sum(C);

M = (2.9/(1+(t/t2)^-2.5) + 1.1) - rho*sum(L);
w_avg = 5*C0;

kfs = as*C.^h./(Ks^h+C.^h);

kfl = al*log(L/L0)./(log(Kl/L0)+log(L/L0));

global aj;
global v_time;

% at each time point, w represents the number of vesicles (spread over
% a Gaussian distribution) arriving in the time step

w = zeros(NN,1);

for i = 1:NN
    for j = 1:size(v_time{i},1)
        w(i) = w(i) + w_avg*aj{i}(j) / (sqrt(2*pi)*sigma)*exp(-(t-
(v_time{i}(j)))^2/(2*sigma^2));
    end
end

% shootin1 concentration in each neurite
dy(1:NN) = -ADV./L.*(C-C0) + w;
% Neurite lengths
dy(NN+1:end) = dkON*M - dkOFF*exp(-kfs + kfl);
```

Appendix B

NeuronGUI, a semi-automated MATLAB interface for quantifying neuronal migration

B.1 NeuronGUI.m

```
% NeuronGUI 2008
% Version 2.1
% Created by Zach Wissner-Gross
% Copyright, 2008

function varargout = NeuronGUI(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @NeuronGUI_OpeningFcn, ...
                   'gui_OutputFcn',  @NeuronGUI_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
function NeuronGUI_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);
if strcmp(get(hObject,'Visible'),'off')
    cla;
    axis equal;
    axis off;
end
function varargout = NeuronGUI_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;
function FileMenu_Callback(hObject, eventdata, handles)
function OpenMenuItem_Callback(hObject, eventdata, handles)
file = uigetfile('*.fig');
if ~isequal(file, 0)
    open(file);
end
function PrintMenuItem_Callback(hObject, eventdata, handles)
printdlg(handles.figure1)
function CloseMenuItem_Callback(hObject, eventdata, handles)
selection = questdlg(['Close ' get(handles.figure1,'Name') '?'],...
                     ['Close ' get(handles.figure1,'Name') '...'],...
                     'Yes','No','Yes');
if strcmp(selection,'No')
    return;
```

```
end
delete(handles.figure1)

function edit_read_Callback(hObject, eventdata, handles)

filename = get(hObject,'String');
handles.edit_read = filename;
guidata(hObject,handles);
cla;

function edit_read_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function read_file_Callback(hObject, eventdata, handles)

filename = handles.edit_read;
set(handles.section_text,'String','Section 0 of ?');
neuron = ParseNeurons(filename);
vertex = ParseVertices(filename);
distance = zeros(0,1);
handles.neuron = neuron;
handles.vertex = vertex;
handles.distance = distance;
section = [0 neuron(end,3)]; % [Current_Section Total_Sections];
handles.section = section;
guidata(hObject,handles);
set(handles.section_text,'String',['Section 0 of '
num2str(section(2))]);

function next_section_Callback(hObject, eventdata, handles)

section = handles.section;
handles.start = zeros(0,2);
handles.finish = zeros(0,2);

if section(1) < section(2)
    section(1) = section(1)+1;
    handles.section = section;
    set(handles.section_text,'String',['Section ' num2str(section(1)) '
of ' num2str(section(2))]);
    CurrentNeuron = SectionNeurons(handles.neuron,section(1));
    CurrentVertex = SectionVertices(CurrentNeuron,handles.vertex);
    handles.current = CurrentNeuron;
    cla;
    hold on;
    plot(CurrentNeuron(:,1),CurrentNeuron(:,2),'b.');
    plot(CurrentVertex(:,1),CurrentVertex(:,2),'w');
    plot(CurrentVertex(:,1),CurrentVertex(:,2),'wo');
    button = 1;
    set(handles.text_wait,'String','Choose start points and then right
click.');
    Start = zeros(0,2);
    while button < 3
        [x,y,button] = ginput(1);
```

```
        if button  == 1
            min = sqrt((CurrentVertex(1,1)-x)^2+(CurrentVertex(1,2)-
y)^2);
            x_min = CurrentVertex(1,1);
            y_min = CurrentVertex(1,2);
            for i = 2:size(CurrentVertex,1)
                if sqrt((CurrentVertex(i,1)-x)^2+(CurrentVertex(i,2)-
y)^2) < min
                    min = sqrt((CurrentVertex(i,1)-
x)^2+(CurrentVertex(i,2)-y)^2);
                    x_min = CurrentVertex(i,1);
                    y_min = CurrentVertex(i,2);
                end
            end
            if size(Start,1) > 0
                if max(sum(ones(size(Start,1),1)*[x_min, y_min] ==
Start,2)) < 2
                    Start = [Start; x_min, y_min];
                    plot(x_min,y_min,'go');
                end
            else
                Start = [Start; x_min, y_min];
                plot(x_min,y_min,'go');
            end
        elseif button == 2
            if size(Start,1) > 0
                plot(Start(end,1),Start(end,2),'wo');
                Start = Start(1:end-1,:);
            end
        end
    end
    button = 1;
    set(handles.text_wait,'String','Choose finish points and then right
click.');
    Finish = zeros(0,2);
    while button < 3
        [x,y,button] = ginput(1);
        if button  == 1
            min = sqrt((CurrentVertex(1,1)-x)^2+(CurrentVertex(1,2)-
y)^2);
            x_min = CurrentVertex(1,1);
            y_min = CurrentVertex(1,2);
            for i = 2:size(CurrentVertex,1)
                if sqrt((CurrentVertex(i,1)-x)^2+(CurrentVertex(i,2)-
y)^2) < min
                    min = sqrt((CurrentVertex(i,1)-
x)^2+(CurrentVertex(i,2)-y)^2);
                    x_min = CurrentVertex(i,1);
                    y_min = CurrentVertex(i,2);
                end
            end
            if size(Finish,1) > 0
                if max(sum(ones(size(Finish,1),1)*[x_min, y_min] ==
Finish,2)) < 2
                    Finish = [Finish; x_min, y_min];
                    plot(x_min,y_min,'ro');
                end
            else
```

```
                Finish = [Finish; x_min, y_min];
                plot(x_min,y_min,'ro');
            end
        elseif button == 2
            if size(Finish,1) > 0
                plot(Finish(end,1),Finish(end,2),'wo');
                Finish = Finish(1:end-1,:);
            end
        end
    end
    if size(Start,1) > 1 && size(Finish,1) > 1
        mid1 = (Start(1,:) + Finish(1,:))/2;
        mid2 = (Start(end,:)+Finish(end,:))/2;
        plot([Start(1,1) mid1(1)],[Start(1,2) mid1(2)],'g');
        plot([Finish(1,1) mid1(1)],[Finish(1,2) mid1(2)],'r');
        plot([Start(end,1) mid2(1)],[Start(end,2) mid2(2)],'g');
        plot([Finish(end,1) mid2(1)],[Finish(end,2) mid2(2)],'r');
    end
    set(handles.text_wait,'String','Press NEXT SECTION.');
end
if section(1) > 0
    CurrentNeuron = handles.current;
    distance = ComputeDistance(CurrentNeuron(:,1:2),Start,Finish);
    handles.distance = [handles.distance; distance];
end
if section(1) == section(2)
    set(handles.text_wait,'String','Write to a file now.');
    figure(2);
    clf;
    hold on;
    hist(handles.distance,[-59.5:1.0:119.5]);
    AX = axis;
    text(AX(1)+0.05*(AX(2)-AX(1)),AX(3)+0.9*(AX(4)-AX(3)),['N = '
num2str(size(handles.distance,1)) ' / '
num2str(size(handles.neuron,1))]);
    text(AX(1)+0.05*(AX(2)-AX(1)),AX(3)+0.8*(AX(4)-AX(3)),['Mean = '
num2str(mean(handles.distance))]);
    text(AX(1)+0.05*(AX(2)-AX(1)),AX(3)+0.7*(AX(4)-AX(3)),['Std Dev = '
num2str(std(handles.distance))]);
    xlabel('Percent migration');
    ylabel('Number of neurons');
    box on;
end

guidata(hObject,handles);

function edit_write_Callback(hObject, eventdata, handles)

filename = get(hObject,'String');
handles.edit_write = filename;
guidata(hObject,handles);

function edit_write_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function write_file_Callback(hObject, eventdata, handles)

distance = handles.distance;
distance = sort(distance);
filename = handles.edit_write;
fid = fopen(filename,'w');
fprintf(fid,'%12.8f\n',distance');
fclose('all');
```

B.2 Parsing scripts

```
% The following code parses the x,y coordinates of all the neurons. The
% third coordinate indicates which section the neurons are in. The
result
% is stored in the array 'neuron.'
function neuron = ParseNeurons(filename)

fid = fopen(filename);
neuron = zeros(0,3);
section = 1;
while 1
    s = fgetl(fid);
    if ~ischar(s)
        break
    elseif findstr(s,'CIRCLE')
        for i = 1:10
            a = fgetl(fid);
            if i == 6
                Ax = str2double(a);
            elseif i == 8
                Ay = str2double(a);
            end
        end
        if size(neuron,1) > 0
            d = sqrt((neuron(end,1)-Ax)^2+(neuron(end,2)-Ay)^2);
            if d > 5*10^3
                section = section + 1;
            end
        end
        neuron = [neuron; Ax Ay section];
    end
end
fclose(fid);

% The following code parses the x,y coordinates of all the vertices.
The
% third coordinate indicates which section the vertices are in. The
result
% is stored in the array 'vertex.'
function vertex = ParseVertices(filename)

fid = fopen(filename);
vertex = zeros(0,2);
while 1
```

```
    s = fgetl(fid);
    if ~ischar(s)
        break
    elseif findstr(s,'VERTEX')
        for i = 1:10
            a = fgetl(fid);
            if i == 4
                Ax = str2double(a);
            elseif i == 6
                Ay = str2double(a);
            end
        end
        if size(vertex,1) > 0
            d = sqrt((vertex(end,1)-Ax)^2+(vertex(end,2)-Ay)^2);
        end
        vertex = [vertex; Ax Ay];
    end
end
fclose(fid);
```

B.3 Ancillary functions

```
function CurrentNeuron = SectionNeurons(neuron,section)

CurrentNeuron = zeros(0,3);
for i = 1:size(neuron,1)
    if neuron(i,3) == section
        CurrentNeuron = [CurrentNeuron; neuron(i,:)];
    end
end

function CurrentVertex = SectionVertices(CurrentNeuron,vertex)

CurrentVertex = zeros(0,2);

% The following code determines the center of mass of the neurons in
the
% current section. The result is stored in 'center.'

sumX = 0;
sumY = 0;
for i = 1:size(CurrentNeuron,1)
    sumX = sumX + CurrentNeuron(i,1);
    sumY = sumY + CurrentNeuron(i,2);
end
center = [sumX sumY]/size(CurrentNeuron,1);

% The following code picks out vertices near the center of mass.

for i = 1:size(vertex,1);
    if sqrt((vertex(i,1)-center(1))^2+(vertex(i,2)-center(2))^2) <
5*10^3
        CurrentVertex = [CurrentVertex; vertex(i,:)];
    end
```

```
end

function distance = ComputeDistance(neuron,start,finish)

if size(start,1) > 1 && size(finish,1) > 1

% The following code weeds out the neurons that are not within the zone
% specified by the start and finish points.

neuron1 = zeros(0,2);
for i = 1:size(neuron)
    minimum = sqrt((start(1,1)-neuron(i,1))^2+(start(1,2)-
neuron(i,2))^2);
    j_min = 1;
    for j = 2:size(start,1)
        d = sqrt((start(j,1)-neuron(i,1))^2+(start(j,2)-
neuron(i,2))^2);
        if d < minimum
            minimum = d;
            j_min = j;
        end
    end
    if j_min == 1
        A = neuron(i,:)-start(1,:);
        B = finish(1,:)-start(1,:);
        C = start(1,:)-start(2,:);
        thetaCA = acos(A*C'/sqrt(A*A'*C*C'));
        thetaCB = acos(B*C'/sqrt(B*B'*C*C'));
        if abs(thetaCA) >= abs(thetaCB)
            neuron1 = [neuron1; neuron(i,:)];
        end
    elseif j_min == size(start,1)
        A = neuron(i,:)-start(j_min,:);
        B = finish(end,:)-start(end,:);
        C = start(end,:)-start(end-1,:);
        thetaCA = acos(A*C'/sqrt(A*A'*C*C'));
        thetaCB = acos(B*C'/sqrt(B*B'*C*C'));
        if abs(thetaCA) >= abs(thetaCB)
            neuron1 = [neuron1; neuron(i,:)];
        end
    else
        neuron1 = [neuron1; neuron(i,:)];
    end
end
neuron = neuron1;
distance = zeros(size(neuron,1),1);

% The following code determines the distance between each neuron and
the
% finish line. Which line segments to use is determined by minimizing
% triangular angles; normal distance is determined using projections.
% The result is stored in 'distance_top.'

distance_top = zeros(size(neuron,1),2);
for i = 1:size(neuron,1)
    minimum = sqrt((finish(1,1)-neuron(i,1))^2+(finish(1,2)-
neuron(i,2))^2);
```

```
    j_min = 1;
    for j = 2:size(finish,1)
        d = sqrt((finish(j,1)-neuron(i,1))^2+(finish(j,2)-
neuron(i,2))^2);
        if d < minimum
            minimum = d;
            j_min = j;
        end
    end
    if j_min == 1
        A = finish(j_min,:)-neuron(i,:);
        C = finish(j_min+1,:)-finish(j_min,:);
        distance_top(i,:) = A - (A*C')/(C*C')*C;
    elseif j_min == size(finish,1)
        A = finish(j_min,:)-neuron(i,:);
        C = finish(j_min-1,:)-finish(j_min,:);
        distance_top(i,:) = A - (A*C')/(C*C')*C;
    else
        B = finish(j_min,:)-neuron(i,:);
        D = finish(j_min,:)-finish(j_min-1,:);
        E = finish(j_min,:)-finish(j_min+1,:);
        theta1 = acos(D*B'/sqrt(D*D'*B*B'));
        theta2 = acos(E*B'/sqrt(E*E'*B*B'));
        if theta1 <= theta2
            if theta1 <= pi/2
                distance_top(i,:) = B-(B*D')/(D*D')*D;
            else
                distance_top(i,:) = B;
            end
        else
            if theta2 <= pi/2
                distance_top(i,:) = B-(B*E')/(E*E')*E;
            else
                distance_top(i,:) = B;
            end
        end
    end
end

% The following code determines the distance between each neuron and
the
% start line. Which line segments to use is determined by minimizing
% triangular angles; normal distance is determined using projections.
The
% result is stored in 'distance_bot.'

distance_bot = zeros(size(neuron,1),2);
for i = 1:size(neuron,1)
    minimum = sqrt((start(1,1)-neuron(i,1))^2+(start(1,2)-
neuron(i,2))^2);
    j_min = 1;
    for j = 2:size(start,1)
        d = sqrt((start(j,1)-neuron(i,1))^2+(start(j,2)-
neuron(i,2))^2);
        if d < minimum
            minimum = d;
            j_min = j;
        end
```

```
    end
    if j_min == 1
        A = start(j_min,:)-neuron(i,:);
        C = start(j_min+1,:)-start(j_min,:);
        distance_bot(i,:) = A - (A*C')/(C*C')*C;
    elseif j_min == size(start,1)
        A = start(j_min,:)-neuron(i,:);
        C = start(j_min-1,:)-start(j_min,:);
        distance_bot(i,:) = A - (A*C')/(C*C')*C;
    else
        B = start(j_min,:)-neuron(i,:);
        D = start(j_min,:)-start(j_min-1,:);
        E = start(j_min,:)-start(j_min+1,:);
        theta1 = acos(D*B'/sqrt(D*D'*B*B'));
        theta2 = acos(E*B'/sqrt(E*E'*B*B'));
        if theta1 <= theta2
            if theta1 <= pi/2
                distance_bot(i,:) = B-(B*D')/(D*D')*D;
            else
                distance_bot(i,:) = B;
            end
        else
            if theta2 <= pi/2
                distance_bot(i,:) = B-(B*E')/(E*E')*E;
            else
                distance_bot(i,:) = B;
            end
        end
    end
end

% The following code determines the percent migration of all the
neurons.
% Whether migration is positive of negative is determined by the
relative
% directionality of distance_top and distance_bot. The result is stored
% in 'distance.' Alas, garbage values are thrown out.

for i = 1:size(neuron,1)
    A = distance_top(i,:);
    B = distance_bot(i,:);
    theta3 = acos(A*B'/sqrt(A*A'*B*B'));
    if theta3 <= pi/2
        distance(i) = 100 * -sqrt(B*B')/(sqrt(A*A')-sqrt(B*B'));
    else
        distance(i) = 100 * sqrt(B*B')/(sqrt(A*A')+sqrt(B*B'));
    end
end

i = 1;
while i < length(distance)+1
    if distance(i) <= -60 || distance(i) >= 120
        if i == 1
            distance = distance(2:end);
        elseif i == length(distance)
            distance = distance(1:end-1);
        else
            distance = [distance(1:i-1);distance(i+1:end)];
```

```
        end
    else
        i = i+1;
    end
end

else
    distance = zeros(0,1);
end
```

References

[1] J. van der Gucht and C. Sykes, Cold Spring Harbor Perspectives in Biology **1**, a001909 (2009).

[2] R. Li and B. Bowerman, Cold Spring Harbor Perspectives in Biology **2**, a003475 (2010).

[3] C. G. Dotti, C. A. Sullivan, and G. A. Banker, Journal of Neuroscience **8**, 1454–1458 (1988).

[4] N. Arimura and K. Kaibuchi, Nature Reviews Neuroscience **8**, 194–205 (2007).

[5] C. G. Dotti and G. A. Banker, Nature **330**, 254–256 (1987).

[6] K. Goslin and G. Banker, The Journal of Cell Biology **108**, 1507–1516 (1989).

[7] K. Khanin and R. Khanin, Journal of Mathematical Biology **42**, 26–40 (2001).

[8] D. C. Samuels, H. G. E. Hentschel, and A. Fine, Philosophical Transactions of the Royal Society of London B **351**, 1147–1156 (1996).

[9] M. Fivaz, S. Bandara, T. Inoue, and T. Meyer, Current Biology **18**, 44–50 (2008).

[10] M. Toriyama, T. Shimada, K. B. Kim, M. Mitsuba, E. Nomura, K. Katsuta, Y. Sakumura, P. Roepstorff, and N. Inagaki, Journal of Cell Biology **175**, 147–157 (2006).

[11] M. Toriyama, Y. Sakumura, T. Shimada, S. Ishii, and N. Inagaki, Molecular Systems Biology **6**, 394 (2010).

[12] N. Inagaki, M. Toriyama, and Y. Sakumura, Developmental Neurobiology **4**, 200–204 (2010).

[13] B. P. Graham and A. van Ooyen, BMC Neuroscience **7**, S9 (2006).

[14] A. van Ooyen, B. P. Graham, and G. J. A. Ramakers, Neurocomputing **38-40**, 73–78 (2001).

[15] M. P. V. Veen and J. Van Pelt, Bulletin of Mathematical Biology **56**, 249–273 (1994).

[16] K. Tsaneva-Atanasova, A. Burgo, T. Galli, and D. Holcman, Biophysical Journal **96**, 840–857 (2009).

[17] G.-H. Li and C.-D. Qin, Mathematical Biosciences **132**, 97–110 (1996).

[18] S. K. W. Dertinger, X. Jiang, Z. Li, V. N. Murthy, and G. M. Whitesides, Proceedings of the National Academy of Sciences **99**, 12542–12547 (2002).

[19] G. N. Li, J. Liu, and D. Hoffman-Kim, Annals of Biomedical Engineering **36**, 889–904 (2008).

[20] W. J. Rosoff, J. S. Urbach, M. A. Esrick, R. G. McAllister, L. J. Richards, and G. J. Goodhill, Nature Neuroscience **7**, 678–682 (2004).

[21] N. E. Sanjana and S. B. Fuller, Journal of Neuroscience Methods **136**, 151–163 (2004).

[22] G. S. Withers, C. D. James, C. E. Kingman, H. G. Craighead, and G. A. Banker, Journal of Neurobiology **66**, (2006).

[23] A. C. von Philipsborn, S. Lang, A. Bernard, J. Loeschinger, C. David, D. Lehnert, M. Bastmeyer, and F. Bonhoeffer, Nature Protocols **1**, 1322–1328 (2006).

[24] A. C. von Philipsborn, S. Lang, J. Loeschinger, A. Bernard, C. David, D. Lehnert, F. Bonhoeffer, and M. Bastmeyer, Development **133**, 2487–2495 (2006).

[25] M. S. Hahn, J. S. Miller, and J. L. West, **18**, 2679–2684 (2006).

[26] K. T. Nguyen and J. L. West, Biomaterials **23**, 4307–4314 (2002).

[27] M. S. Hahn, J. S. Miller, and J. L. West, Advanced Materials **17**, 2939–2942 (2005).

[28] Y. Luo and M. S. Shoichet, Nature Materials **3**, 249–253 (2004).

[29] T. A. Kapur and M. S. Shoichet, **68A**, 235–243 (2004).

[30] B. Kaehr, R. Allen, D. J. Javier, J. Currie, and J. B. Shear, Proceedings of the National Academy of Sciences **101**, 16104–16108 (2004).

[31] J. M. Belisle, J. P. Correia, P. W. Wiseman, Ti. E. Kennedy, and S. Costantino, Lab on a Chip **8**, 2164–2167 (2008).

[32] M. A. Holden and P. S. Cremer, Journal of the American Chemical Society **125**, 8074–8075 (2003).

[33] M. A. Scott, Z. D. Wissner-Gross, and M. F. Yanik, Lab on a Chip **12**, 2265–76 (2012).

[34] C. Xu and W. W. Webb, Journal of the Optical Society of America B **13**, 481–491 (1996).

[35] P. Shi, M. a Scott, B. Ghosh, D. Wan, Z. Wissner-Gross, R. Mazitschek, S. J. Haggarty, and M. F. Yanik, Nature Communications **2**, 510 (2011).

[36] V. J. Peschansky, T. J. Burbridge, A. J. Volz, C. Fiondella, Z. Wissner-Gross, A. M. Galaburda, J. J. Lo Turco, and G. D. Rosen, Cerebral Cortex **20**, 884–97 (2010).

[37] T. J. Burbridge, Y. Wang, a J. Volz, V. J. Peschansky, L. Lisann, a M. Galaburda, J. J. Lo Turco, and G. D. Rosen, Neuroscience **152**, 723–33 (2008).

[38] Z. D. Wissner-Gross, M. A. Scott, D. Ku, P. Ramaswamy, and M. Fatih Yanik, Integrative Biology **3**, 65–74 (2011).

[39] T. C. Südhof, Nature **455**, 903–911 (2008).

[40] D. H. Geschwind and P. Levitt, Current Opinion in Neurobiology **17**, 103–111 (2007).

[41] A. Bellon, Molecular Psychiatry **12**, 620–629 (2007).

[42] A. J. Larner, Developmental Cell **4**, 249–258 (1995).

[43] S. S. L. Andersen and G. Bi, Bio Essays **22**, 172–179 (2000).

[44] C. Menager, N. Arimura, Y. Fukata, and K. Kaibuchi, Journal of Neurochemistry **89**, 109–118 (2004).

[45] P. Lamoureux, R. E. Buxbaum, and S. R. Heidemann, Journal of Cell Science **111**, 3245–3252 (1998).

[46] T. Esch, V. Lemmon, and G. Banker, Journal of Neuroscience **19**, 6417–6426 (1999).

[47] T. J. Diefenbach, P. B. Guthrie, and S. B. Kater, Journal of Neuroscience **20**, 1484–1494 (2000).

[48] L. M. Y. Yu, J. H. Wosnick, and M. S. Shoichet, Journal of Neuroscience Methods **171**, 253–263 (2008).

[49] J. Q. Zheng, Nature **403**, 89–93 (2000).

[50] A. A. Oliva, C. D. James, C. E. Kingman, H. G. Craighead, and G. A. Banker, Neurochemistry Research **28**, 1639–1648 (2003).

[51] B. C. Wheeler, J. M. Corey, G. J. Brewer, and D. W. Branch, Journal of Biomechanical Engineering **121**, 73–78 (1999).

[52] A. Welle, S. Horn, J. Schimmelpfeng, and D. Kalka, Journal of Neuroscience Methods **142**, 243–250 (2005).

[53] P. Shi, K. Shen, and L. C. Kam, Developmental Neurobiology **67**, 1765–1776 (2007).

[54] P. Clark, S. Britland, and P. Connolly, Journal of Cell Science **105**, 203–212 (1993).

[55] D. R. McLean and B. P. Graham, Mathematical Medicine and Biology **23**, 101–117 (2006).

[56] D. R. McLean and B. P. Graham, Proceedings: Mathematical, Physical and Engineering Sciences **460**, 2437–2456 (2004).

[57] H. C. Berg and D. A. Brown, Nature **239**, 500–504 (1972).

[58] T. Mitchison and M. Kirschner, Nature **312**, 237–242 (1984).

[59] O. Feinerman, A. Rotem, and E. Moses, Nature Physics **4**, 967–97 (2008).

[60] Z. Wissner-Gross, M. Scott, J. Steinmeyer, and F. Yanik, PLoS ONE **Accepted**, (2012).

[61] G. A. Banker and W. M. Cowan, Brain Research **126**, 397–342 (1977).

[62] A. Matus, R. Bernhardt, R. Bodmer, and D. Alaimo, Neuroscience **17**, 371–389 (1986).

[63] T. Nishimura, T. Yamaguchi, K. Kato, M. Yoshizawa, Y.-I. Nabeshima, S. Ohno, M. Hoshino, and K. Kaibuchi, Nature Cell Biology **7**, 270–277 (2005).

[64] T. Mori, T. Wada, T. Suzuki, Y. Kubota, and N. Inagaki, The Journal of Biological Chemistry **282**, 19884–19893 (2007).

[65] W. Yu and P. W. Baas, Journal of Neuroscience **14**, 2818–2829 (1994).

[66] M. Kempf, A. Clement, A. Faissner, G. Lee, and R. Brandt, The Journal of Neuroscience **16**, 5583–92 (1996).